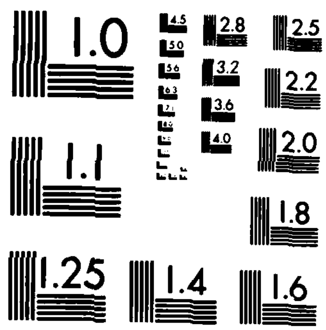
AD-A151 094 PARTS REQUIREMENTS AND COST MODEL (PARCOM) DOCUMENTATION PARCOM FUNCTIONAL DESCRIPTION(U) ARMY CONCEPTS ANALYSIS AGENCY BETHESDA MD W J BAUMAN OCT 84 1/1

UNCLASSIFIED CAA-D-84-15 F/G 15/5 NL

END

FILMED

DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A151 094

CAA-D-84-15

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER CAA-D-84-15	2. GOVT ACCESSION NO. ADF 860019	3. RECIPIENT'S CATALOG NUMBER AD-A151094
4. TITLE (and Subtitle) Parts Requirements and Costs Model (PARCOM) Documentation - PARCOM Functional Description		5. TYPE OF REPORT & PERIOD COVERED Final Report May 84 - Oct 84
		6. PERFORMING ORG. REPORT NUMBER CAA-D-84-15
7. AUTHOR(s) Mr. Walter J. Bauman		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS US Army Concepts Analysis Agency 8120 Woodmont Avenue Bethesda, Maryland 20814-2797		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE October 1984
		13. NUMBER OF PAGES 76
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release. Distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

The work reported was performed for:
Office of the Deputy Chief of Staff for Logistics
Department of the Army

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

PARCOM Model, logistics model, spare parts, spares, stockage model, inventory management, aircraft, helicopters, logistics requirements, readiness, functional description.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Documentation was developed for the Parts Requirements and Costs Model (PARCOM) developed at the US Army Concepts Analysis Agency. PARCOM was designed to provide the Army with an analytical tool for quick reaction, gross estimation of wartime spare parts requirements and costs as they relate to flying hour and availability objectives. An ability to identify problem parts and possible causes of the problems was also desired. The PARCOM Functional Description is structured to provide a user with detailed

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CAA-D-84-15

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Block 20 - ABSTRACT Continued

information on PARCOM model logic, restrictions and potential for extension. Additional information on model application may be found in the PARCOM User's Guide, published separately.

DTIC COPY INSPECTED 1

Accession For	
NTIS GRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

DOCUMENTATION
CAA-D-84-15

PARTS REQUIREMENTS AND COST MODEL (PARCOM) DOCUMENTATION

PARCOM FUNCTIONAL DESCRIPTION

OCTOBER 1984

PREPARED BY
FORCE SYSTEMS DIRECTORATE
US ARMY CONCEPTS ANALYSIS AGENCY
8120 WOODMONT AVENUE
BETHESDA, MARYLAND 20814-2797

CONTENTS

FIGURES

TABLES

TABLE		Page
2-10	Unconstrained Cost Total Requirement (Initial Inventory = 0) with "NMCS = 0"	2-14
2-11	Unconstrained Cost Residual Requirement with "NMCS = 0"	2-14
2-12	Unconstrained Cost Total Requirement (Initial Inventory = 0) with No Substitution - Part 1 Requirement Calculation	2-15
2-13	Unconstrained Cost Total Requirement (Initial Inventory = 0) with No Substitution - Part 2 Requirement Calculation	2-16
2-14	Unconstrained Cost Residual Requirement with No Substitution - Part 1 Requirement Calculation	2-17
2-15	Unconstrained Cost Residual Requirement with No Substitution - Part 2 Requirement Calculation	2-17
2-16	Constrained Cost Residual Requirement with No Substitution - Cost Limit = $4,300	2-18
2-17	Constrained Cost Total Requirements with No Substitution - Cost Limit = $104,800	2-18
2-18	Total Requirement (Initial Inventory = 0) Summary	2-19
2-19	Residual (add-on) Requirements Summary	2-19
2-20	Capability Assessment for Unconstrained Cost Total Requirement - Full Substitution	2-20
2-21	Capability Assessment for Unconstrained Cost Total Requirement - No Substitution	2-21
2-22	Assessment Iterations for Constrained Cost Residual Requirement	2-24
2-23	Capability Assessment for Constrained Cost Residual Requirement	2-24
4-1	PARCOM Arrays with a Day Limit Dimension	4-3
4-2	PARCOM Arrays with a Parts Limit Dimension	4-4

PARTS REQUIREMENTS AND COST MODEL (PARCOM) DOCUMENTATION

PARCOM FUNCTIONAL DESCRIPTION

CHAPTER 1

GENERAL DESCRIPTION

1-1. PURPOSE OF THE FUNCTIONAL DESCRIPTION. This functional description of the Parts Requirements and Cost Model (PARCOM) provides:

a. The structure of the model logic which will serve as a basis for mutual understanding between the user and the developer.

b. Information on model restrictions, potential for extension, and user impacts.

1-2. PROJECT REFERENCES

a. Parts Requirements and Cost Model (PARCOM) User's Guide, CAA-D-84-10, US Army Concepts Analysis Agency, October 1984.

b. Aircraft Spare Stockage Methodology (Aircraft Spares) Study, CAA-SR-84-12, US Army Concepts Analysis Agency, April 1984.

c. Pickard, W. C., Zellner, P. A., and Bailey, D. R., DOD Assimilation of US Air Force Methodologies for Relating Logistics Resources to Materiel Readiness, SYNERGY, Inc., August 1983.

1-3. TERMS AND ABBREVIATIONS. The reader is directed to the glossary at the end of this document.

1-4. DEVELOPMENT BACKGROUND. The US Army Concepts Analysis Agency (CAA) developed PARCOM, a model for generating cost effective mixes of aircraft spare parts which satisfy specified scenario conditions. Development occurred during the course of the Aircraft Spare Stockage Methodology (Aircraft Spares) Study conducted by CAA. That study, and PARCOM development, were in response to interest shown by the Deputy Chief of Staff for Logistics (DCSLOG) in developing a methodology (or methodologies) relating aircraft spare parts stockage levels to combat readiness and flying hour capability. The calculation of spare parts requirements and of the effects of budgeting changes has been a slow and cumbersome peacetime-oriented exercise. The principal criterion for spares stockage has been the achievement of acceptable stockout, or fill rate, levels. To more realistically predict wartime spare parts requirements, and to better justify budget requests for spare parts procurement, the Army needed a more responsive methodology based on wartime flying hour expectations and system readiness/availability requirements.

1-5. STRUCTURE OF ARMY AIRCRAFT LOGISTICS

a. Governing Regulations. Policy and procedural guidance for the Army's inventory management efforts is largely contained in two regulations:

- AR 710-1 Centralized Inventory Management of the Army Supply System
- AR 710-2 Supply Policy Below the Wholesale Level

(1) AR 710-1 establishes responsibilities and procedures for centralized inventory management of Army materiel by the major subordinate commands (MSC) of the US Army Materiel Command (AMC).

(2) AR 710-2 prescribes supply procedures to be used at the retail level, including methods for determining authorized stockage lists and appropriate stockage levels.

b. Maintenance System Structure. Figure 1-1 illustrates the interaction of supply, maintenance, and industrial activities within the aircraft parts logistics system.

(1) Parts Storage Locations. Aircraft spare parts are stored with using units at the Aviation Unit Maintenance (AVUM) and the Aviation Intermediate Maintenance (AVIM) levels. Aircraft spare parts are stored in various CONUS depots for shipment to users upon requisition. Additionally, war reserve parts are stored in various CONUS depots or prepositioned in the appropriate theater.

(2) Participating Organizations and Responsibilities. AVUM facilities are organic to the lower echelon aviation units which actually fly and maintain the Army's aircraft. These user units stock a prescribed load list (PLL) of repair parts at the AVUM level. PLLs are sized to sustain the unit's anticipated wartime flight operations for a specified number of days (usually 15). Stockage levels and reordering procedures are governed by AR 710-2. AVIM units develop their own authorized stockage lists (ASL) based on demands for parts received from supported AVUM units and from their own AVIM operations. AVIM ASLs are exclusive of subordinate unit PSLs. The development of ASLs is also governed by AR 710-2. Part types are selected for PLL and ASL stockage based upon a combination of experienced demand frequency and mission essentiality. The AVIM/AVUM (retail) parts requirements are supported by stocks maintained in supply depots (wholesale) in CONUS. Automated inventory management techniques are employed by AVSCOM to authorize and record fill of retail requisitions by the appropriate wholesale depot. Depot stocks are replenished through procurement of new parts or repair of returned unserviceables.

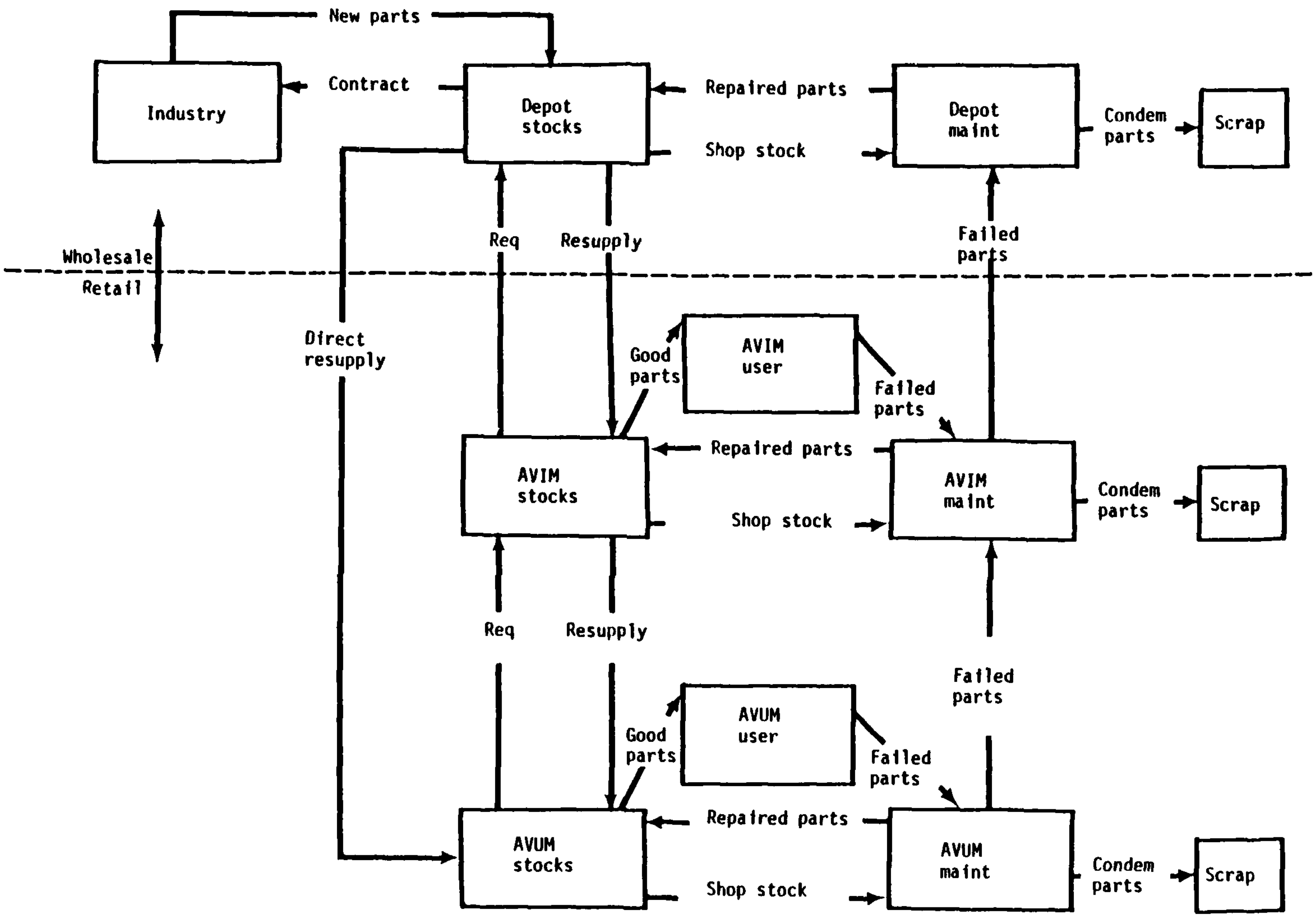


Figure 1-1. Aircraft Parts L tics System

c. **Areas of Consideration**

(1) **Peacetime versus Wartime.** Peacetime requirements for spare parts are computed based upon experienced annual demand and projected peacetime usage. AVSCOM uses an automated system of data bases and models to forecast these requirements, and bases its computations on a supply availability goal. Wartime requirements are computed and funded separately from peacetime requirements, and address those parts required to sustain the force during the initial stages of war until lines of communication and supply can be established. The primary consideration for peacetime requirements is meeting supply availability goals, while that for war reserve requirements is meeting sustainability goals.

(2) **Initial Provisioning versus Replenishment.** Computation of the spare parts requirement for initial provisioning of new weapons systems is necessarily based on less concrete data than is that for replenishment parts for already fielded systems. No demand history has yet been developed, so engineering estimates of parts failure factors are used instead. In many cases, all the parts to be included in the new aircraft have not been fully identified, and their cost must be extrapolated from that of a list of major assemblies. AVSCOM has an automated capability to compute initial provisioning requirements based on these projected data. Over the first 2 years of a system's life, actual demand data is accumulated and given increasing weight in spare parts management decisions. After a system has been fielded for 2 years, its replenishment spare parts requirements are computed using actual demand data to the maximum extent possible.

(3) **Retail versus Wholesale.** The Army splits its inventory management into "retail" and "wholesale" activities. In the aviation logistics context, AVUM- and AVIM-level parts stockages are termed "retail," while those at the depot level are termed "wholesale." The methodologies used to compute spare parts requirements for the retail and wholesale levels are entirely different and essentially unrelated. Retail stockage levels are computed and authorized based upon a combination of demand experience, combat essentiality, and mobility requirements. AR 710-2 establishes computational procedures used by retail parts managers to determine their stockage levels and appropriate reorder points. Wholesale parts requirements are computed based upon average monthly demand experienced at the wholesale level. Wholesale item managers have little visibility of retail spare parts postures or weapons system availabilities. Rather, wholesale parts are procured or repaired at rates calculated to achieve a chosen demand satisfaction percentage without backorders.

(4) **Fill Rate versus System Availability Criteria.** AVSCOM computes spare parts requirements with the objective of achieving a target fill rate. Its goal is to fill a selected percentage of all demands received without having to backorder parts. The item manager does not base his parts management decisions on weapons system availability, and in fact, has little or no visibility of this retail level criterion. Department of Defense (DOD)

deployed. Computed "cumulative aircraft surviving" entries are defined by the difference between "cumulative aircraft deployed" and "cumulative aircraft lost." Since, for simplicity, our example shows a zero aircraft attrition rate, surviving aircraft is equal to deployed aircraft. The "program flying hours" column gives the flying hour objective in terms of required program flying hours for the fleet on each day. The last column gives the availability objective in terms of an input-specified daily minimum (fleet) aircraft availability required each day. The input-specified "maximum flying hours per aircraft per day" is also noted at the bottom of the table.

Table 2-4. Scenario Data

Day	Cumulative aircraft deployed	Cumulative aircraft lost	Cumulative aircraft surviving	Program flying hours	Minimum aircraft availability
1	150	0	150	500	.10
2	200	0	200	1,000	.09
3	200	0	200	1,000	.09
4	200	0	200	1,500	.09
5	200	0	200	1,500	.09

Maximum flying hours per aircraft per day = 10.
Cost limit for constrained cost case = $4,300.
Desired convergence (constrained cost) = 0.
Maximum iterations (constrained cost) = 2.

c. **Calculation of Daily Allowable NMCS Aircraft.** Table 2-5 shows results of the calculation of allowable NMCS aircraft for each day. Each result in the rightmost column is the surviving aircraft minus the larger of:

(1) The minimum aircraft required to achieve the daily flying hour objective, for each day, computed as "program flying hours" divided by "maximum flying hours per aircraft per day."

(2) The minimum aircraft required to achieve the daily availability objective, for each day, computed as the product of "surviving aircraft" and "minimum aircraft availability."

Component calculations for the first day, using the data of Table 2-4, are shown.

2-2. EXAMPLE. The algorithm logic described in the previous paragraph can be better understood through use of a manual example. The tables to follow portray a stylized but useful hypothetical example which uses only "back-of-the envelope" calculations. The tables all apply to one case and are presented in the same sequence as the model algorithms described in the previous paragraph.

a. **Part Data Base.** Tables 2-2 and 2-3 show a part data base for two part types. Recall that failure rate is in terms of failures per flying hour and QPA = number of parts installed per operational aircraft.

Table 2-2. Part Characteristic Data

Part	Failure rate	QPA	Unit cost	Initial inventory
1	.08	1	$400	250
2	.02	1	$ 50	10

The time units in Table 2-3 are in days. The last column of Table 2-3 is the computed repair cycle calculated from the other data in that row, e.g., for Part 1, the repair cycle = 2 * OST + depot repair time = 3 days. The repair cycle for a part is defined as the average time between failure of a part and its (repaired) return to the retail spare pool. Only the repair cycle entry will be used in succeeding calculations because it includes the effects of the other data in Table 2-3.

Table 2-3. Part Repair Time Data

Part	OST	Depot repair time	Retail repair time	NRTS fraction	Depot condemned	Retail condemned	Repair cycle
1	1	1	0	1.00	0	0	3
2	0	0	3	.00	0	0	3

b. **Scenario Data Base.** Table 2-4 shows the scenario data for the case. A 5-day "war" is shown. The aircraft status (deployed, lost) entries are for the start of the associated day of the war. Thus, for example, 50 aircraft are newly deployed at the start of day 2. By "cumulative aircraft deployed" is meant all aircraft deployed in theater from the start of the war through the given day. No aircraft are assumed withdrawn once

g. Capability Assessment of Constrained Cost Requirements Mix. PARCOM also generates the daily fleet availability and flying hour capability achieved with constrained cost solution mixes. Recall that these mixes are derived for a "no substitution" policy only. With unconstrained costs, net demand was based on the entire planned flying hour program being flown. For a constrained cost mix, some unknown (at first) number of hours will be flown. That number must initially be estimated and an iterative approach, as shown in Figure 2-6, applied to determine NMCS aircraft, availability, and achievable program flying hours. For each day, therefore, a starting estimate of flying hours flown is made (the first day's starting estimate is the program flying hours). Then, net demand, as based on the estimated flying hours, is computed, followed by implied NMCS aircraft (generated by the estimated flying hours), achievable flying hours, and flying hours per available aircraft. The achievable flying hours are compared with the estimated flying hours flown. If, based on input thresholds, they are close enough, the iterations stop. If not, the calculations are repeated based on a new starting estimate of flying hours equal to the average of the estimated and the achieved flying hours. After iterations for a day are completed, the available aircraft for the day and their flying hour potential are calculated based on the last calculation of NMCS aircraft and on the maximum flying hour potential per aircraft per day (an input). Processing for the next day uses a starting estimate of flying hours based on the "achieved flying hours" of the previous day.

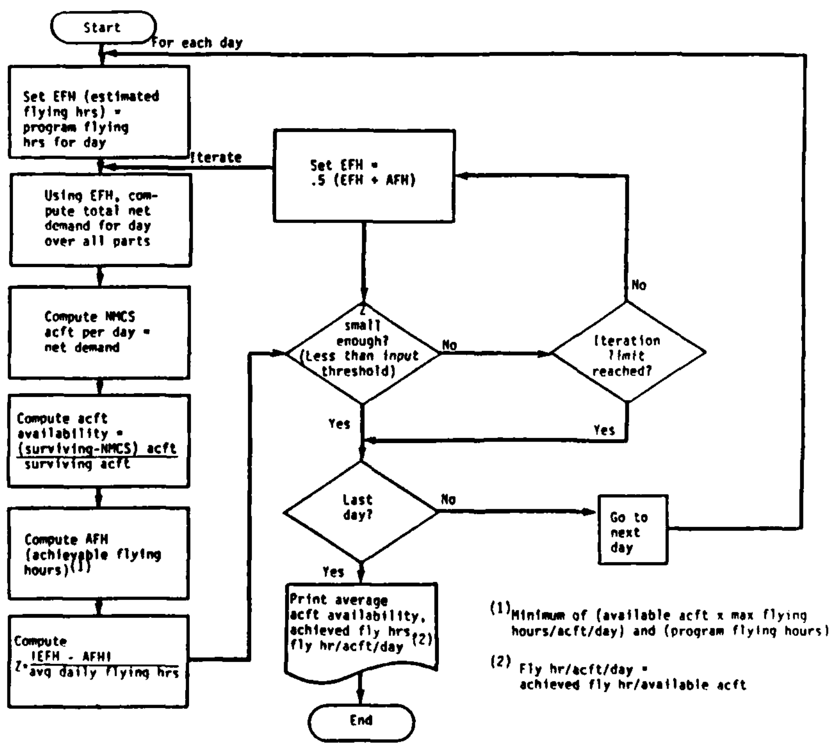


Figure 2-6. PARCOM Computation Algorithm for Constrained Cost Capability Assessment

initial inventory is assumed to be the sum of the computed requirement and the original initial inventory. For each computed unconstrained cost requirements mix, PARCOM generates a record of achieved daily and average aircraft availability, achieved program flying hours, and achieved flying hours per available aircraft per day. The achieved program flying hours are simply the program flying hours, by definition, of an unconstrained cost solution. Also by definition, aircraft availability = 1.00 for a "NMCS = 0" policy. The PARCOM capability assessment algorithm assesses daily and average availability for both the full substitution" and "no substitution" policies. The calculations depend principally on the net demand and NMCS determinations explained earlier. Recall that for a "no substitution" policy, each stockout creates an NMCS aircraft, so the sum of stockouts over all parts is also the number of NMCS aircraft created. For a "full substitution" policy a single NMCS aircraft may have stockouts for several different parts. In this case the number of NMCS aircraft created is the largest value, over all parts, of the quotient of stockouts divided by QPA for each part type. For each day, the number of NMCS aircraft is subtracted from the number of surviving aircraft to yield available aircraft. Availability is then the ratio of available to surviving aircraft. Flying hours per available aircraft is just the daily program flying hours divided by the number of available aircraft for the day.

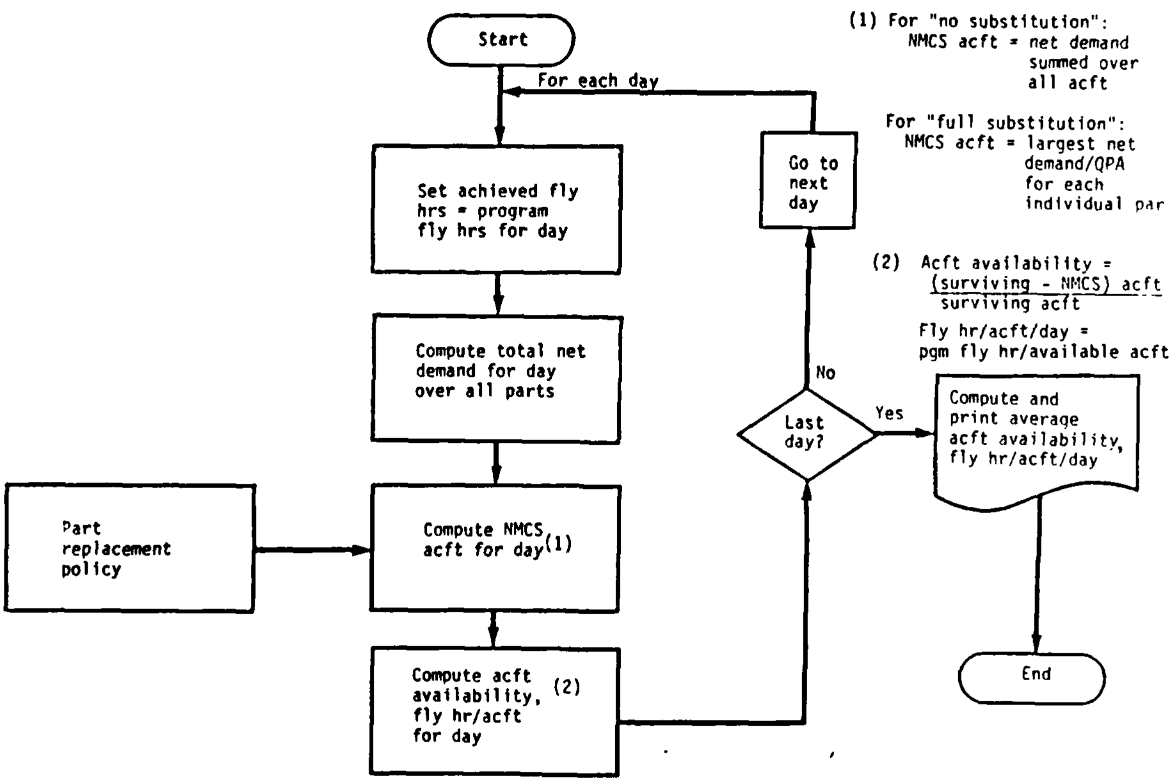


Figure 2-5. PARCOM Computation Algorithm for Unconstrained Cost Capability Assessment

item (regardless of part type) creates an NMCS aircraft. Therefore, our constrained cost solution mix minimizes the instances of NMCS created by the constrained funds. The solution tends, heuristically, toward the achievement of maximum cumulative flying hours.

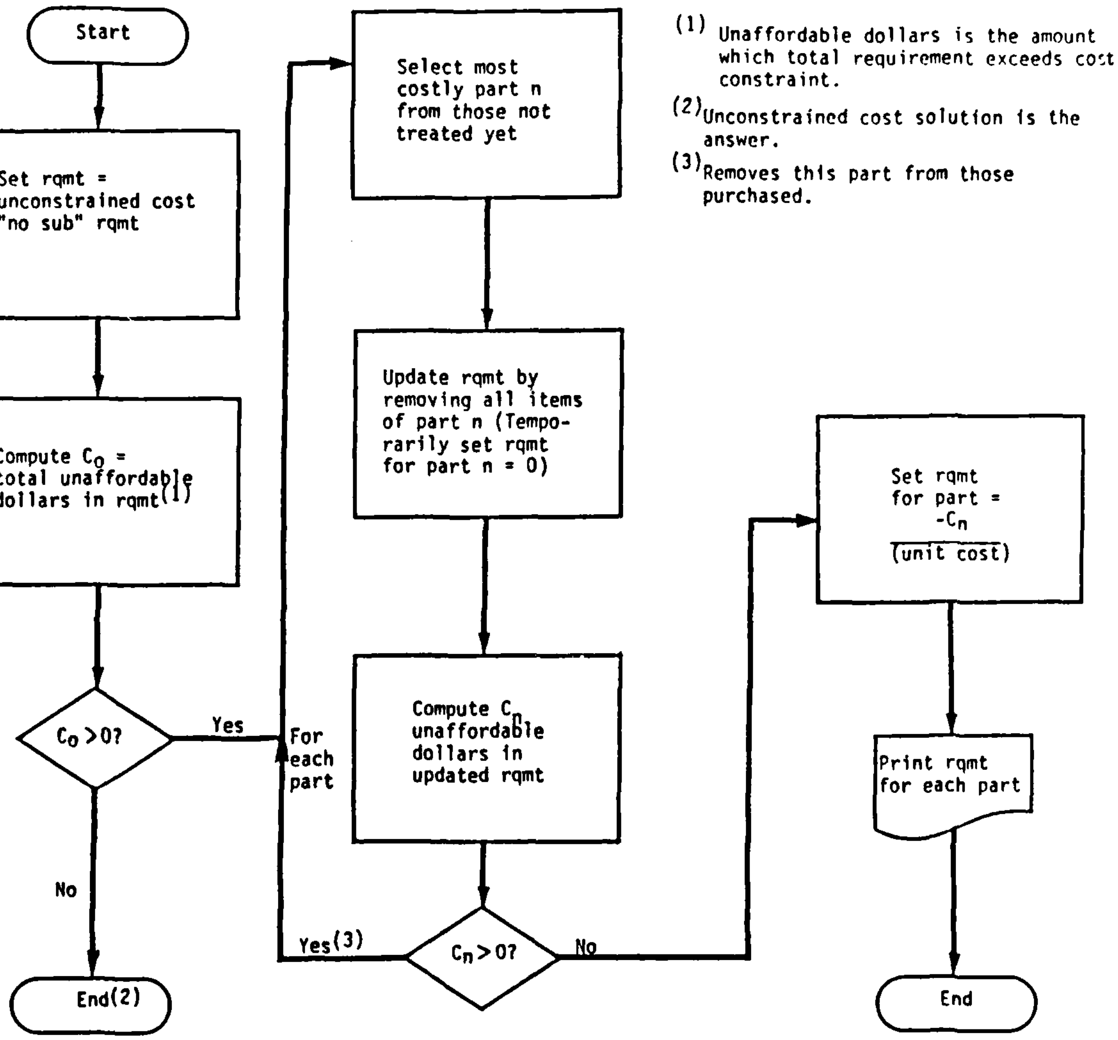


Figure 2-4. PARCOM Requirements Computation Algorithm for Constrained Cost with "No Substitution"

f. Capability Assessment of Unconstrained Cost Requirements Mix. Figure 2-5 illustrates the PARCOM computation algorithm for capability assessment of the unconstrained cost requirements solutions. After each unconstrained cost solution mix for the "no substitution" and "full substitution" cases is computed, PARCOM generates a record of daily and average fleet operational capability achievable by stocking each computed requirement, i.e., the new

(2) With "no substitution," PARCOM determines allowed stockout and net demand for the most expensive parts first. Allowed stockout is, again, the number of permissible NMCS aircraft. Because there is no overlap/consolidation of stockout effects (as was the case for "full substitution"), requirements computations for parts are interdependent. An iterative algorithm is used to reflect part interdependence. The algorithm of Figure 2-3 also applies to the "no substitution" and "NMCS = 0" requirements. The points affected by policy differences during algorithm application are summarized in Table 2-1. Understanding of calculation of "no substitution" parts requirements is assisted by reference to the example of paragraph 2-2.

Table 2-1. Differences in Application of PARCOM Unconstrained Cost Requirements Algorithm by Policy

Policy	Algorithm procedure/calculation	
	Allowable stockout	Order of processing
Full sub	Allowed NMCS acft x QPA	Irrevelant
No sub	Allowed NMCS acft	By decreasing part cost
NMCS = 0	0	Irrelevant

e. Constrained Cost "No Substitution" Requirement. After the unconstrained cost "no substitution" requirements are computed, they become the basis for the constrained cost solution. A cost limit on spares is input along with the other scenario and objective data. A constrained cost parts mix can be constructed by the simulated "spending" of money to "buy", in order of increasing part unit cost, the part requirements of the unconstrained cost solution until the money is exhausted. That would entail the procurement of the largest number of total parts from the unconstrained cost solution. However, another characteristic of such a constrained cost parts mix is that it is the mix which has the fewest "unbought" (hence, unstocked) items from the unconstrained cost solution. The PARCOM algorithm, shown in Figure 2-4, arrives at its solution by calculating "unbought" items. Initially, it "spends" the full cost of the unconstrained cost requirements mix assuming it to be the constrained cost solution. PARCOM subsequently selects the fewest number of items to remove from that solution until the remaining parts mix is priced at the input cost limit. Because the programed algorithm solves by "unbuying" items rather than "buying" them, parts are processed in decreasing order of part unit cost. Notice that under a policy of "no substitution" each "unbought"

(1) Net demand (for all three replacement policies) for a part at any point in time is the cumulative removals to that time minus the sum of cumulative returning repairs and initial inventory. Removals are generated by the product of failure rate, part QPA (quantity installed per aircraft), and programed flying hours. Returning repairs are generated by removed parts cycling through a "repair pipeline" and being returned to the point of removal. A positive net demand represents a shortage of the part.

(2) Under "full substitution" the aircraft frames providing the sources of parts substituted for failed parts when spares are unavailable are consolidated to the minimum possible number, i.e, there will be a maximum overlap of aircraft frames providing missing parts. Because of this overlap, the spare parts requirements for each part may be independently computed. For a "full substitution" policy, the allowable stockout for a part on any day is the product of allowable NMCS aircraft for that day and the part QPA.

(3) As indicated by Figure 2-3, the minimum spare requirement for a part needed to achieve the case objective on any day is the net demand for that part minus the allowable stockout. The overall spare requirement for a part is the largest of the daily minimum spare requirements for that part. It is a least cost solution because it is the smallest purchase of that part which will permit the case objective to be met on all days.

c. Unconstrained Cost "NMCS = 0" Requirement. The "NMCS = 0" policy corresponds to the case in which 100 percent aircraft availability is required every day. In such a case allowed NMCS aircraft and allowable stockout both must be zero every day. The "NMCS = 0" case could be considered a special case of a "full substitution" case with a 100 percent aircraft availability objective (the "no substitution" case with that objective would yield the same answer, because part substitution policy is irrelevant when no stockouts are allowed). The spares required by the solution to the "NMCS = 0" case also can be interpreted as the total expected net demand for a part during the war. It is a least cost solution because any amount less than that required to meet the expected demand will create an NMCS aircraft, i.e., will not meet the case objective.

d. Unconstrained Cost "No Substitution" Requirement

(1) Under "no substitution," the stockouts generated by parts removals in excess of on-hand spares must each be associated with separate aircraft frames. Every missing part results in an inoperable (NMCS) aircraft. It is most cost effective, therefore, to assign the allowed stockout (allowed number of NMCS aircraft) to the most expensive parts. For example, if 50 aircraft are allowed to be NMCS and a shortage exists of 50 expensive parts and 50 cheap ones, the 50 cheap ones need to be bought. If 75 expensive parts and 50 cheaps ones are short, there will be no choice but to buy 25 expensive ones (leaving 50 unbought) and 50 cheap ones, in order to best meet the case objective.

b. Unconstrained Cost "Full Substitution" Requirement. Figure 2-3 shows the PARCOM algorithm used to compute a requirements solution for all three parts replacement policies with unconstrained costs. The difference between "full substitution" and "no substitution" calculations is in the ways that allowed stockouts are calculated. Net demand is the same for each.

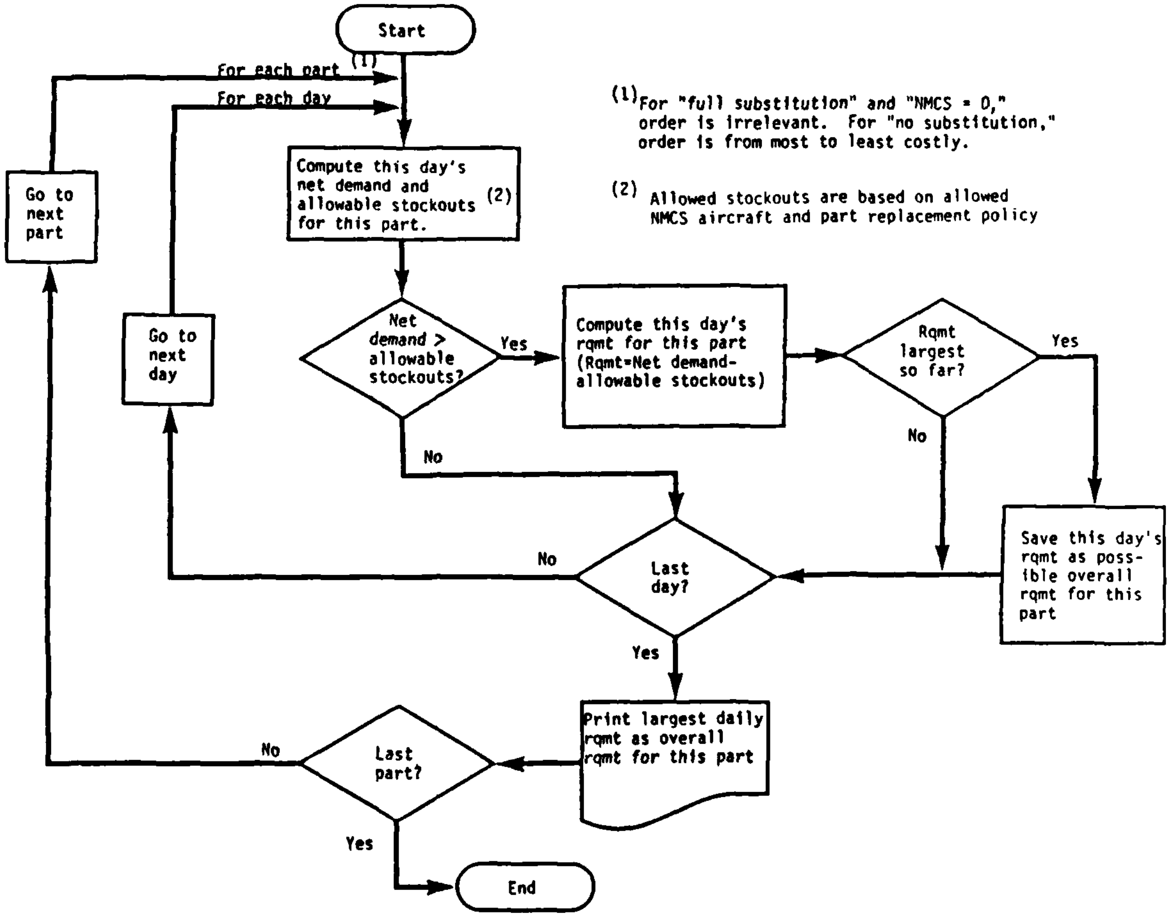


Figure 2-3. PARCOM Requirements Computation Algorithm for Unconstrained Costs, "Full Substitution", "No Substitution", and "NMCS = 0"

a. **Calculation of Daily Allowable NMCS Aircraft.** To meet flying hour and availability goals, the maximum number of aircraft allowed to be down due to a lack of parts (allowable NMCS aircraft) is determined for each day. As shown in Figure 2-2, separate minimums are computed of aircraft required to meet the flying objective and the availability objective (if any). The largest of the two minimums is subtracted from the number of surviving aircraft on each day to yield the "allowable NMCS aircraft" for that day. Within the subsequent processing algorithms, the "allowable NMCS aircraft" is converted to an "allowable stockout" for each part and replacement policy. The "allowable stockout" for a part on a day is just the maximum number of backorders (unfilled demands) for the part which will still allow accomplishment of the case objective (flying hour and availability) on that day, i.e., these are parts that are missing but which don't have to be bought.

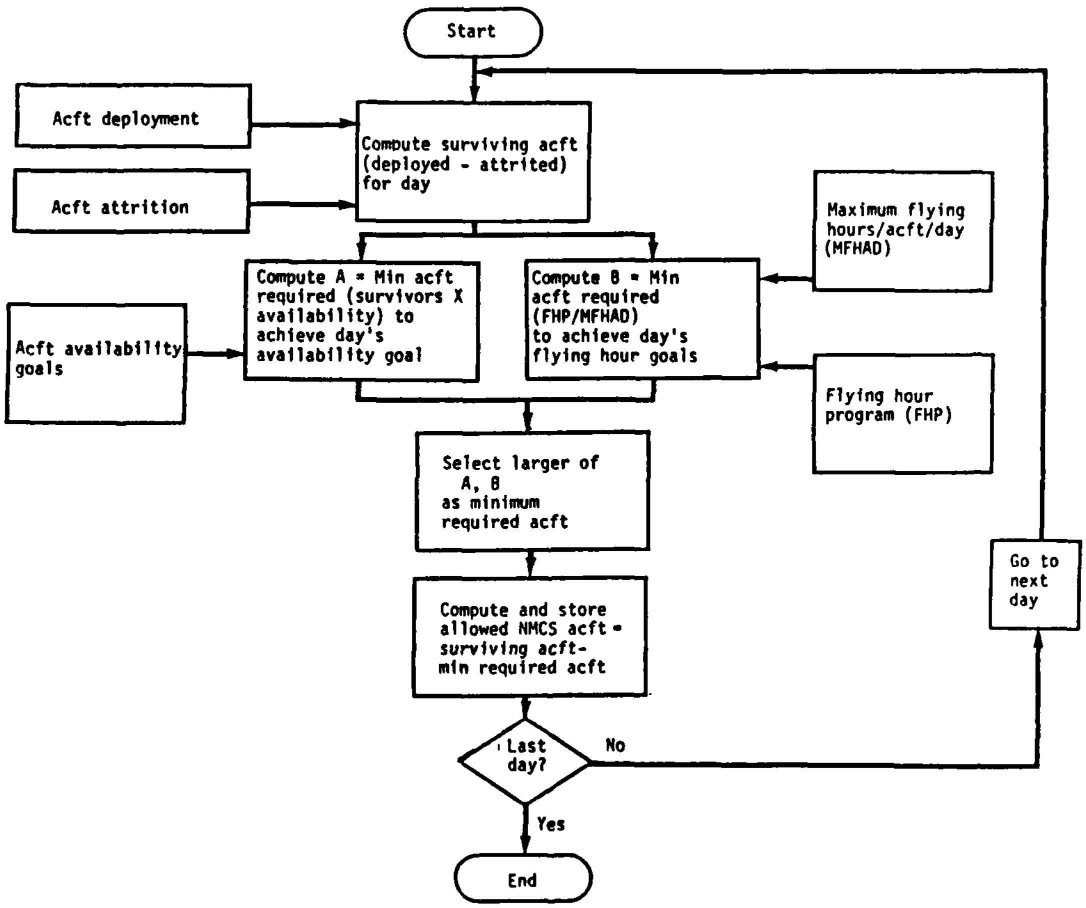


Figure 2-2. PARCOM Computation Algorithm for Allowable NMCS Aircraft

CHAPTER 2

PARCOM LOGIC

2-1. ALGORITHMS. PARCOM is a series of expected value simulations of the spare part requirements generation process for cases defined by a combination of parameters noted in the previous chapter. In addition, the model computes the capability potential of the force when operated with each computed spares mix. The assessed capability potential is in terms of achievable aircraft availability and fraction of the flying hour program which can be accomplished. Figure 2-1 illustrates the general nature and sequence of PARCOM processing. The basic model sequence, with logic diagrams as appropriate, is described as follows:

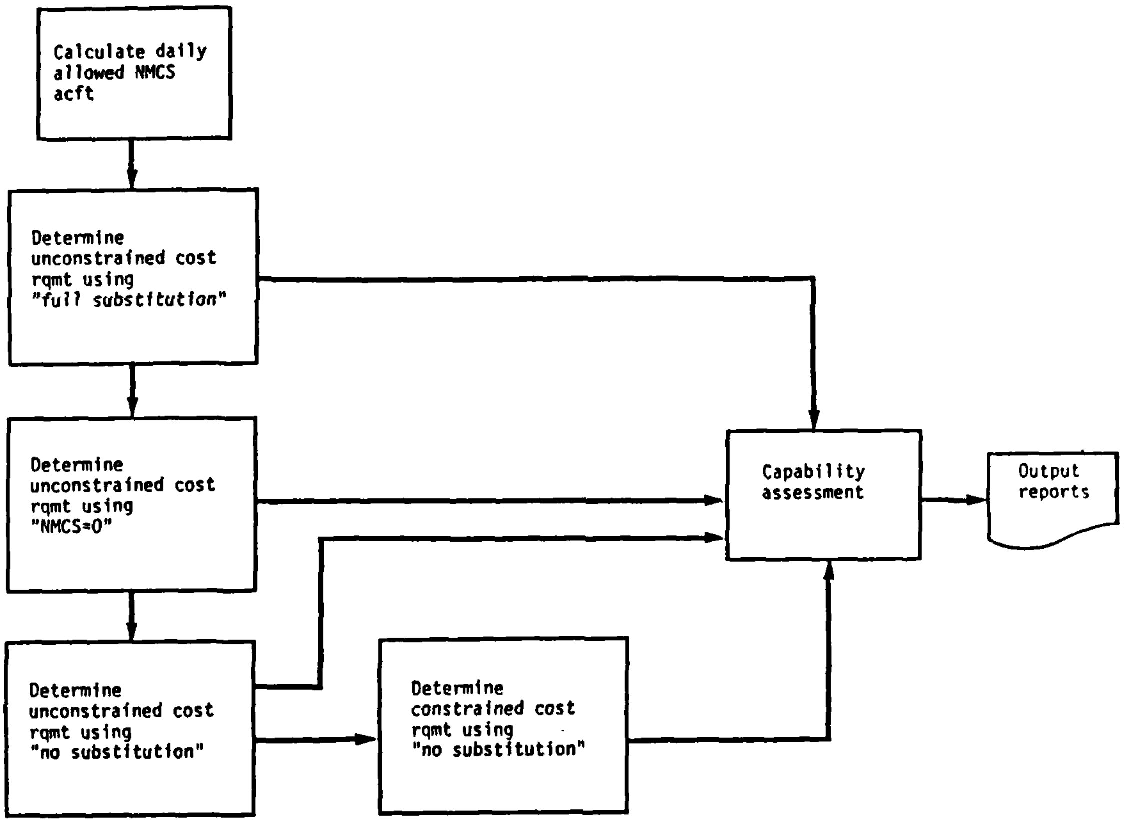


Figure 2-1. PARCOM Processing Sequence

(2) **Residual Requirement.** Best add-on (to input initial inventory) requirements mix, with a "no substitution" policy, that can be bought with a funding limit equal to the input cost limit.

(3) **Daily Aircraft Available.** For each day of the full scenario, the fraction of surviving aircraft which are not NMCS, assuming that the initial spare inventory is set equal to the sum of the computed parts requirement and the original initial inventory.

(4) **Daily Flying Hour Fraction.** For each day of the full scenario, the fraction of the fleet flying program which can be achieved assuming that the initial spare inventory is set equal to the sum of the computed parts requirement and the original initial inventory.

(5) **Daily Flying Hours per Aircraft per Day.** For each day of the scenario, the average achieved flying hours per aircraft per day, assuming the computed solution parts mix is stocked.

1-9. TYPICAL PROBLEMS ADDRESSED. A single PARCOM run can provide answers to several problems pertinent to a given scenario. From the user point of view, typical problem statements, given a specified aircraft deployment schedule, flying program, and attrition scenario are:

a. What is the least cost add-on buy needed to achieve the flying program using "full substitution" parts replacement and requiring that NMCS not exceed 0.15 on all days? What is the associated daily NMCS status?

b. With a budget limit of $10,000,000, what spares should be added to current inventory, using a "no substitution" policy, to increase, to the extent possible, the fraction of the flying program achieved? What is the associated daily NMCS status? What is the associated fraction of the flying program that is achievable?

d. Aircraft Availability Objective. An aircraft availability objective is a requirement for a specific minimum aircraft availability on each day (different days may have different minimum required availabilities). In this context, aircraft availability = 1 - NMCS, where NMCS = the fraction of surviving aircraft in "not mission capable supply" status. An aircraft is in an NMCS status if it is nonoperational because spare parts are needed but are not available to restore it to serviceability. Specification of availability objectives is in addition to the flying hour objective. Specification of a zero availability objective is equivalent to no availability objective at all.

1-8. SUMMARY OF PARCOM OUTPUT. The following are the basic types of print output produced by PARCOM for requirements problems. Details may be found in the PARCOM User's Guide.

a. Unconstrained Cost Cases

(1) Total Requirement. Total least-cost parts mix and costs required to achieve the case objectives (flying program and availability) given a zero initial inventory.

(2) Residual Requirement. The least-cost add-on parts mix (to an input initial inventory) and costs required to achieve the case objectives.

(3) Cumulative Cost by Day. For each day N (N=1, 2,..., through end of "war"), the total and the add-on cost of the full parts requirement to meet the case objectives through day N only, i.e., it is the cost of the requirement for a truncated scenario of N days. Parts mix is not shown.

(4) Cumulative Requirement by Day. For selected items, for each day N, the cumulative total requirement needed (in the full parts scenario) to meet the case objectives through N days. A zero initial inventory is assumed in this output.

(5) Daily Aircraft Available. For each day of the full scenario, the fraction of surviving aircraft which are not NMCS, assuming that the initial spare inventory is set equal to the sum of the computed parts requirement and this original initial inventory.

(6) Daily Flying Hours per Aircraft per Day. For each day of the scenario, the average achieved flying hours per available aircraft per day assuming the computed solution parts mix is stocked.

b. Constrained Costs

(1) Total Requirement. Total "best" requirements mix, with zero initial inventory, and with a "no substitution" policy, that can be bought with a funding limit equal to the sum of the value of "refunded" current spares inventory and the input cost limit. The objective of a "best" mix is to maximize flying hour productivity with the constrained funds.

failures. Gross part failures interact with initial spare inventory and the repair process at depot and at retail to produce a net demand for spare parts at user level. The net demand for spare parts at user level then determines the number of surviving aircraft that are mission capable or not mission capable supply (NMCS). As will be seen in the next chapter, PARCOM simulates all interactions in expected value terms, i.e., in terms of the product of an average process rate and the number of "items" subjected to that process.

1-7. PARCOM PROBLEM SPECIFICATION. The basic purpose of PARCOM is to generate cost-effective mixes of add-on spare parts needed to permit an aircraft fleet of specified type to achieve specified flying program and availability goals under various cost constraints, part replacement policies, and aircraft availability objectives. These are described below in summary fashion. Additional detail may be found in the PARCOM User's Guide.

a. **Cost Constraints.** The two cost constraint modes are:

(1) **Unconstrained Funds** - where unlimited funds for procurement of additional required parts are assumed available.

(2) **Constrained Funds** - where a cost (funding) limit for add-on spares is set. If unable to meet the flying hour and, possibly, availability objectives with the limited funds, the model generates a "best" solution mix with the funds available, i.e., it seeks to maximize program flying hours achievable within the funding constraint.

b. **Part Replacement Policies.** The two basic part replacement policies are:*

(1) **Full Substitution** - where a failed part on an aircraft may be replaced by either a spare (if available) or by a serviceable part from a "not mission capable" (NMC) aircraft (if a spare is not available).

(2) **No Substitution** - where a failed part on an aircraft may only be replaced by a spare part. With constrained funds, PARCOM operates in this mode only.

c. **Flying Hour Objective.** A flying hour objective is a requirement for the aircraft fleet to achieve a specified number of flying hours on each day of the scenario. An input flying hour program designates the daily goal. A basic PARCOM objective is to generate a parts mix which will achieve the specified flying program at least cost.

---

*"NMCS = 0" is treated as a third part replacement policy but is really a special case of "no substitution" in which aircraft availability is constrained to be 100 percent.

has expressed its support for implementation of system availability-driven parts requirements computation methodologies in all the armed services. The primary difficulty for the Army is the collection of accurate data to drive such automated models.

d. **Similarity of Aircraft and Other Spares Procurement.** Each of the MSCs uses the Commodity Command Standard System (CCSS) to meet its inventory management responsibilities. The processes used are essentially the same for all types of spares.

1-6. PARCOM REPRESENTATION OF LOGISTICS ENVIRONMENT. The PARCOM "world view" of the aircraft part logistics system is based on the representation in Figure 1-1. PARCOM, however, has only two echelons of stock and repair, viz.:

a. **Wholesale Level.** This level consists of the "depot stocks" and "depot maintenance" blocks of Figure 1-1. Depot maintenance is represented in terms of depot repair times, depot condemnation rates, and order ship times (OST) between depot and retail level. PARCOM assumes that initial spares stockage at depot can be made available to retail level before users at retail have "drawn down" initial in-theater stocks, i.e., efficient transportation and positioning of initial spares stock is assumed. The effect of this assumption is the same as that resulting from front-loading of initial stocks at retail level. Production of new parts by industry is not treated in PARCOM.

b. **Retail Level.** This level is treated as one pool (or "bin") of spare parts stocks consisting of all stocks at AVIM and AVUM levels in Figure 1-1. Retail maintenance is treated as an aggregate process and is represented in terms of retail repair times, not repairable this station (NRTS) percentages, and retail condemnation rates. Essentially, "retail" represents pooled AVIM and AVUM functions.

c. **Users.** Users of spare parts are deployed aircraft. PARCOM treats deployed aircraft only at retail level. These are augmented by (input) scheduled deployments of additional aircraft (from a presumed rear area) during the course of a simulated "war." Currently, PARCOM can treat only a homogeneous aircraft fleet of one type for a single force. Deployed aircraft are subject to attrition based on (input) attrition factors. Combat is not explicitly represented.

d. **Failure Generation.** The deployed aircraft fleet is assigned (via input) a flying hour program, broken into daily flying hour requirements. PARCOM finds a cost-effective mix of spare parts, which, over the course of the "war", will, on average, achieve the set flying program in addition to specified daily aircraft availability requirements. If spares procurement funds are constrained, PARCOM seeks a cost-effective spares mix achieving as much of the flying program as possible. Input failure rates for spare parts are in terms of failures per flying hour. In general, achieved flying hours interact with part failure rates to produce gross part

Table 2-5. Calculation of Allowable NMCS Aircraft

Day	Minimum aircraft required		Allowable NMCS acft
	Flying hour objective	Availability objective	
1	500/10 = 50	150*.10 = 15	150-50 = 100
2	100	18	100
3	100	18	100
4	150	18	50
5	150	18	50

d. **Unconstrained Cost Full Substitution Total Requirement.** Tables 2-6 and 2-7 show the calculations for the total requirement (initial inventory = 0) under full substitution. Each "cumulative net demand" entry is just the "cumulative failures" minus the sum of the "cumulative returning repairs" and the initial inventory. "Cumulative failures" is based on the program hours being flown and is computed by accumulating (over days) the product of failure rate, QPA, and program flying hours for each day (as taken from Tables 2-3 and 2-4). Initial inventory is set to zero as prescribed for the calculation of "total requirements." The "cumulative returning repairs" entries are the "cumulative failures" entries lagged by three days (the repair cycle from Table 2-3). Any condemnations (our case has none) would have to be deducted from the lagged failures. If R is the length of the repair cycle for a part (see Table 2-3), PARCOM treats all noncondemned failures occurring at the start of day n as being returned to the retail spare pool at the start of day n + R. If a part has both a depot repair cycle and a retail repair cycle, PARCOM would partition repairs over the two cycles. In our simplified example, Part 1 has only a depot repair cycle of three days while Part 2 has only a retail repair cycle of three days. The "allowable stockouts" under full substitution is calculated as the product of "allowable NMCS aircraft" (from Table 2-5) and the part QPA. Since QPA = 1 for both parts (see Table 2-2), allowable stockouts = allowable NMCS aircraft. The "day requirement" is calculated as the larger of zero and (cumulative net demand minus allowable stockouts). The overall requirement for each part is determined as the largest value (over days) of the "day requirement" entries. It is circled in each table. Component calculations are displayed for the first day based on the data of Tables 2-2 through 2-5. At the end of Table 2-7, the total cost of the computed requirements mix is also shown based on the unit cost data in Table 2-2.

Table 2-6. Unconstrained Cost Total Requirement (Initial Inventory = 0) with Full Substitution - Part 1

Day	Cumulative failures	Cumulative returning repairs	Cumulative net demand	Allowable stockouts	Day rqmt
1	.08*500 = 40	0	40-0-0 = 40	100	max(0,40-60) = 0
2	120	0	120	100	20
3	200	0	200	100	100
4	320	40	280	50	230
5	440	120	320	50	270

Table 2-7. Unconstrained Cost Total Requirement (Initial Inventory = 0) with Full Substitution - Part 2

Day	Cumulative failures	Cumulative returning repairs	Cumulative net demand	Allowable stockout	Day rqmt
1	.02*500 = 100	0	10-0-0 = 10	100	max(0,10-100) = 0
2	30	0	30	100	0
3	50	0	50	100	0
4	80	10	70	50	20
5	110	30	80	50	30

Total requirements cost = Part 1 rqmt * 400 + Part 2 rqmt * 50
= 270 * 400 + 30 * 50 = $109,500.

e. **Unconstrained Cost Full Substitution Residual Requirement.** Tables 2-8 and 2-9 show calculations for the residual requirement under full substitution. The principal difference is that initial inventory = 250 for Part 1 and = 10 for Part 2. The "cumulative net demand" column entries are less than the previous case because the nonzero initial inventory is subtracted from the previous values. The logic is the same as in the previous case. The computed overall requirement (circled) is, now, just the add-on requirement to the specified initial inventory. The total cost also reflects only the add-on requirement.

Table 2-8. Unconstrained Cost Residual Requirement (Initial Inventory = 250) with Full Substitution - Part 1

Day	Cumulative failures	Cumulative returning repairs	Cumulative net demand	Allowable stockout	Day rqmt
1	.08*500 = 40	0	max(0,-210) = 0	100	max(0,0-100) = 0
2	120	0	0	100	0
3	200	0	0	100	0
4	320	40	30	50	0
5	440	120	70	50	20

Table 2-9. Unconstrained Cost Residual Requirement (Initial Inventory = 10) with Full Substitution - Part 2

Day	Cumulative failures	Cumulative returning repairs	Cumulative net demand	Allowable stockout	Day rqmt
1	.02*500 = 10	0	10-0-10 = 0	100	max(0,0-100) = 0
2	30	0	20	100	0
3	50	0	40	100	0
4	80	10	60	50	10
5	110	30	70	50	20

Total residual requirements = Part 1 rqmt * 400 + Part 2 rqmt * 50 = 20 * 400 + 20 *50 = $9,000.

f. **Unconstrained Cost "NMCS = 0" Total Requirement.** As noted previously, this is a special case of the unconstrained cost "full substitution" calculations in which we set "allowable stockouts = 0". Table 2-10 shows the calculations. "Cumulative net demand" is the same as in Tables 2-6 and 2-7 because it is not affected by part replacement policy. As a general rule, the "day requirement" under "NMCS = 0" is the same as "cumulative net demand" for that day. The overall requirement for each part is circled in the figure. Total costs are given below the figure.

Table 2-10. Unconstrained Cost Total Requirement (Initial Inventory = 0) with "NMCS = 0"

Day	Part 1			Part 2		
	Cumlative net demand	Allowable stockouts	Day rqmt	Cumulative net demand	Allowable stockouts	Day rqmt
1	40	0	40-0 = 40	10	0	10-0 = 10
2	120	0	120	30	0	30
3	200	0	200	50	0	50
4	280	0	280	70	0	70
5	320	0	320	80	0	80

Total requirement cost = 320 * 400 + 80 * 50 = $132,000.

g. **Unconstrained Cost "NMCS = 0" Residual Requirement.** Table 2-11 shows calculations for the residual requirement under "NMCS = 0". The "cumulative net demand" entries are the same as in Tables 2-8 and 2-9. The same logic used in the "full substitution" case applies here. The computed overall requirement (circled) is only the add-on to the input-specified initial inventory.

Table 2-11. Unconstrained Cost Residual Requirement with "NMCS = 0"

Day	Part 1 (init inv = 250)			Part 2 (init inv = 10)		
	Cumulative net demand	Allowable stockouts	Day rqmt	Cumulative net demand	Allowable stockouts	Day rqmt
1	0	0	0	0	0	0
2	0	0	0	20	0	20
3	0	0	0	40	0	40
5	30	0	30	60	0	60
5	70	0	70	70	0	70

Total residual requirement = 70 * 400 + 70 * 50 = $31,500.

h. Unconstrained Cost "No Substitution" Total Requirement. Tables 2-12 and 2-13 show calculations for our example problem. The tables are presented in the required sequence of computations, i.e., the most expensive part (Part 1) is processed first. The "cumulative net demand" is the same as used in Table 2-6 because it is not altered by the substitution policy used. The "day requirement" is just the cumulative net demand (the shortage on that day) minus the allowable stockout (the allowed shortage) for that day (but not less than zero). The overall Part 1 requirement is the circled largest "day requirement". The Part 1 requirement is treated as "purchased" during further processing (for other part requirements). Table 2-13 shows the calculation of the second part requirement which must be for the next most expensive part (i.e., Part 2 in our example). The "purchase" of the Part 1 requirement alters the initial inventory for that part. Therefore, the old cumulative net demand for Part 1 (based on initial inventory = 0) in Table 2-12 is reduced by 270 (the new initial inventory for Part 1) to generate the new cumulative net demand for that part. The new cumulative net demand for Part 1 is just the number of stockouts which must be allocated from that part. For a "no substitution" policy the total allowed stockout consists of the summed stockouts over all parts treated. For each day, the cumulative net demand for Part 1 acts as a "lock" or "claimant" on the same number of stockouts in the original allowable stockout. Requirements for Part 2 can only be based on the unallocated allowable stockout, tabulated in Table 2-13, which is the original allowed stockout minus all "claimant" stockouts (net demands) from parts already processed. Since the Part 2 requirement is not yet "purchased" (it is being computed), the "cumulative net demand" column of Table 2-7 is repeated in Table 2-13. The "day requirement" in Table 2-13 is calculated as the cumulative net demand for Part 2 minus the unallocated allowable stockout. As before, the overall requirement (circled) is the largest of the day requirements. The Part 2 requirement would be assumed "purchased," and the process would be continued with less expensive parts (if any). Each successive calculation would use an "unallocated allowable stockout" equal to the original (Table 2-12) allowable stockout reduced by the sum total of allocated stockouts reflected in "purchases" of parts already processed.

Table 2-12. Unconstrained Cost Total Requirement (Initial Inventory = 0) with No Substitution - Part 1 Requirement Calculation

Day	Cumulative net demand	Allowable stockouts	Day rqmt
1	40	100	0
2	120	100	20
3	200	100	100
4	280	50	230
5	320	50	270

Table 2-13. Unconstrained Cost Total Requirement (Initial Inventory = 0) with No Substitution - Part 2 Requirement Calculation

Day	Cumulative net demand		Unallocated allowable stockouts	Day rqmt
	Part 1 (init inv = 270)	Part 2 (init inv = 0)		
1	0	10	100-0 = 100	0
2	0	30	100-0 = 100	0
3	0	50	100-0 = 100	0
4	280-270 = 10	70	50-10 = 40	70-40 = 30
5	320-270 = 50	80	50-50 = 0	80-0 = 80

Total requirements cost = 270 * 400 + 80 * 50 = $112,000.

i. **Unconstrained Cost "No Substitution" Residual Requirement.** Tables 2-14 and 2-15 show calculations for our example problem. The basic logic is exactly the same as for the previous case (initial inventory = 0). However, the residual requirements and the net demand used to compute them are based on the input-specified initial inventory (Part 1 = 250, Part 2 = 10). The "cumulative net demand" in Table 2-14 is the same as the "cumulative net demand" in Table 2-8 (because it depends only on initial inventory, not substitution policy). Part 1 requirements are calculated in the same manner as in the previous case. The calculations for the second part (Part 2) in Table 2-15 differ from those in Table 2-13 only because "cumulative net demand" for Part 2 is taken from Table 2-9, i.e., is based on initial inventory = 10. The "cumulative net demand" entries for Part 1 in Table 2-15 are the same as those in Table 2-13 because the "purchase" of 20 of Part 1 (the computed requirement in Table 2-13), when added to an initial inventory of 250, yields the same "new initial inventory" as the purchase of 270 of Part 1 (the computed requirement of Table 2-11) added to a zero initial inventory. The requirements for this case are add-on requirements.

Table 2-14. Unconstrained Cost Residual Requirement with No Substitution - Part 1 Requirement Calculation (Initial Inventory = 250)

Day	Cumulative net demand	Allowable stockouts	Day rqmt
1	0	100	0
2	0	100	0
3	0	100	0
4	30	50	0
5	70	50	20

Table 2-15. Unconstrained Cost Residual Requirement with No Substitution - Part 2 Requirement Calculation

Day	Cumulative net demand		Unallocated allowable stockouts	Day rqmt
	Part 1 (init inv = 270)	Part 2 (init inv = 10)		
1	0	0	100-0 = 100	0
2	0	20	100-0 = 100	0
3	0	40	100-0 = 100	0
4	280-270 = 10	60	50-10 = 40	60-40 = 20
5	320-270 = 50	70	50-50 = 0	70-0 = 70

Total residual requirements cost = 20 * 400 + 70 * 50 = $11,500.

j. **Constrained Cost Add-on Requirements.** Table 2-16 shows calculations for the constrained cost add-on requirements with "no substitution" and with an add-on cost limit of $4,300. The table is arranged in steps. In step 1 the constrained cost requirement is (temporarily) set equal to the unconstrained cost no substitution residual requirement computed in Tables 2-14 and 2-15. The "rqmt cost" column shows the cost of that requirement. The "unaffordable dollars" column shows the amount by which that requirement cost exceeds our specified cost limit. Since that entry is positive, the requirement is updated by selecting the most expensive part from those with nonzero add-on requirements (Part 1 in our case) and by setting the requirement for that part to zero. The new requirement is shown in step 2. Since the cost of the new requirement is less than the cost limit, the "unaffordable dollars" are negative. In that case the final requirements are set by updating the requirement for the last part processed (i.e., the

part with its requirement set to zero) to the quotient of the last unaffordable dollars entry and unit cost = (-800)/400 = 2 in our example. The final add-on requirements are circled in Table 2-16.

Table 2-16. Constrained Cost Residual Requirement with No Substitution - Cost Limit = $4,300

Step	Requirement (init inv = 250, 10)		Rqmt cost	Unaffordable dollars
	Part 1	Part 2		
1	20	70	$11,500	$7,200
2	0	70	$ 3,500	$ -800
3	2	70	$ 4,300	0

k. **Constrained Cost Total Requirement.** Table 2-17 shows calculations for this case, which is based on a zero initial inventory and a cost limit equal to the cost of current inventory (250 * 400 + 10 * 50 = 100,500) plus the input cost limit (4,300). The resulting cost limit is $104,800. The logic is the same as in the previous case. Only the numbers are changed to reflect the zero initial inventory base. Thus, the step 1 cost requirement is the unconstrained cost total requirement from Tables 2-12 and 2-13. Note that the final requirement is equivalent to the sum of the residual requirement computed in Table 2-16 and the input initial inventory. Such equivalence is not always true.

Table 2-17. Constrained Cost Total Requirement with No Substitution - Cost Limit = $104,800

Step	Requirement (init inv = 0)		Rqmt cost	Unaffordable dollars
	Part 1	Part 2		
1	270	80	$ 112,000	$ 7,200
2	0	80	$ 4,000	$-100,800
3	252	80	$ 104,800	0

l. **Requirements Summary.** Table 2-18 presents a summary of all total requirement cases (initial inventory = 0) treated thus far. Table 2-19 summarizes the residual (add-on) requirement cases.

Table 2-18. Total Requirements (Initial Inventory = 0) Summary

Cost constraint	Requirements (init inv = 0)		Rqmt cost
	Part 1	Part 2	
Unconstrained cost			
Full sub	270	30	$109,500
No sub	270	80	$112,000
NMCS = 0	320	80	$132,000
Constrained cost (no sub) (Limit = $104,800)	252	80	$104,800

Table 2-19. Residual (Add-on) Requirements Summary

Cost constraint	Requirements (init inv = 250, 10)		Rqmt cost
	Part 1	Part 2	
Unconstrained cost			
Full sub	20	20	$ 9,000
No sub	20	70	$11,500
NMCS = 0	70	70	$31,500
Constrained cost (no sub) (Limit = $4,300)	2	70	$ 4,300

m. **Capability Assessment of Unconstrained Cost Total Requirements - Full Substitution.** Table 2-20 shows the capability assessment calculations of the expected effects of stocking the requirements computed in Tables 2-6 and 2-7. Cumulative net demand for each part type is based on initial inventories being set to the computed requirements. "NMCS aircraft" for each day are set equal to the larger of the (cumulative net demand) QPA entries for the day. "Surviving aircraft" are as in Table 2-4. "Available aircraft" are "surviving aircraft" minus "NMCS aircraft." Aircraft availability is the quotient of available and surviving aircraft. Flying hours per (available) aircraft per day are calculated by dividing the program flying hours for each day (see Table 2-4) by the number of available aircraft on that day. Average availability is constructed by weighting daily availabilities by the daily surviving aircraft. Average flying hours per (available) aircraft per day is weighted by the available aircraft on each day.

Table 2-20. Capability Assessment for Unconstrained Cost Total Requirement - Full Substitution

Day	Cumulative net demand/QPA		NMCS acft	Surviving acft	Available acft	Aircraft availability	Flying hrs per acft per day
	Part 1 (init inv = 270)	Part 2 (init inv = 30)					
1	0	0	0	150	150	1.00	3.3
2	0	0	0	200	200	1.00	5.0
3	0	(50-30)/1	20	200	130	180/200 = .90	5.6
4	(280-270)/1	(70-30)/1	40	200	160	160/200 = .30	9.4
5	(320-270)/1	(80-30)/1	50	200	150	150/200 = .75	10.0

Avg availability = [(150 * 1) + (200 * 1) + (200 * .9) + (200 * .8) + (200 * .75)] /950 = .88

Avg flying hrs/acft/day = [(150 * 3.3) + (200 * 5) + (180 * 5.6) + (160 * 9.4) + (150 * 10)] /840 = 6.6

n. Capability Assessment of Unconstrained Cost Residual Requirements - Full Substitution. Table 2-20 also applies to this case because the residual requirements (calculated in Tables 2-8 and 2-9), when "stocked" and added to the input-specified initial inventory, yield the same new initial inventory as resulted from stocking the total (initial inventory = 0) requirements.

o. Capability Assessment of Unconstrained Cost Total Requirement - No Substitution. Table 2-21 shows the capability assessment calculations for the expected effects of stocking the requirements computed in Tables 2-12 and 2-13. Cumulative net demand for each part type is based on initial inventories being set to the computed requirements. Under a "no substitution" policy, "NMCS aircraft" for each day are then equal to the sum of the "cumulative net demands" entries for that day. "Surviving aircraft" are as in Table 2-4. Component calculations are displayed.

Table 2-21. Capability Assessment for Unconstrained Cost Total Requirement - No Substitution

Day	Cumulative Net Demand		NMCS acft	Surviving acft	Aircraft availability	Flying hours per aircraft per day
	Part 1 (init inv = 270)	Part 2 (init inv = 80)				
1	0	0	0	150	1.00	500/150 = 3.3
2	0	0	0	200	1.00	1,000/200 = 5.0
3	0	0	0	200	1.00	5.0
4	280-270 = 10	0	10	200	190/200 = .95	1,500/190 = 7.9
5	320-270 = 50	80-80 = 0	50	200	150/200 = .75	1,500/150 = 10

Avg availability = [(150 x 1.) + (200 x 1.) + (200 x 1.) + (200 x .95) + (200 x .75)]/890 = 6.2

Avg flying hrs/acft/day = [(150 x 3.3) + (200 x 5.) + (200 x 5.) + (190 x 7.9) + (150 x 10.)]/890 = 6.2

p. Capability Assessment of Unconstrained Cost Residual Requirement- No Substitution. Table 2-21 also applies to this case because the residual requirements (calculated in Tables 2-14 and 2-15), when "stocked" and added to the input-specified initial inventory, yield the same new initial inventory as resulted from stocking the total requirements (initial inventory = 0).

q. Capability Assessment of Constrained Cost Residual (Add-on) Requirement. Tables 2-22 and 2-23 show the capability assessment calculations for the expected effects of stocking the requirements computed in Table 2-16. The scenario data (Table 2-4) specified a limit of two iterations in seeking convergence of "estimated flying hours" to "achieved flying hours" in the calculations. Each day starts (step 1) with an "estimated flying hours" based on the program flying hours for that day. Cumulative net demands are then generated for each part type, based on the estimated flying hours, resulting in NMCS aircraft as shown. The available aircraft then determine "achieved program flying hours". If the "achieved flying hours" are close enough (based on specified input convergence requirement) to the "estimated flying hours", the day's calculations are done. If not, a new step is made with a new flying hour estimate until either desired convergence is attained or the input-specified number of steps (iterations) have been performed on that day. For the first four days the achieved flying hours equal the estimated flying hours so only one iteration is required.

(1) By way of illustration, the entries for day 4 in Table 2-22 are calculated as follows:

- Flying hour estimate = 1,500 = program hours.
- Part 1 cumulative net demand (init inv = 252) = the largest of cumulative net demand from Table 2-8 (init inv = 250) - 2 = 30 -2 = 28 or 0 = 0.
- Part 2 cumulative net demand (init inv = 80) = the larger of cumulative net demand from Table 2-9 (init inv = 10) - 70 = 60 - 70 = -10 or 0 = 0
- NMCS aircraft = sum of (2) and (3) = 28.
- Available aircraft = surviving acft - NMCS acft = 200 - 28 = 172.
- Achieved flying hours = available acft * max fly hr/acft/day or the program hours for the day (the smaller is chosen) = (172 * 10 or 1,500) = 1,500.

(2) On day 5, there is no convergence, as is shown in the calculations for step 1 shown below:

- Flying hour estimate = 1,500 = program hours.
- Part 1 cumulative net demand (init inv = 252) = cumulative failures from (500 + 1,000 + 1,000 + 1,500 + 1,500) flying hours -cumulative returned repairs - 252 = .08 * (5,500) - 120 - 252 = 68.
- In a similar way, Part 2 cumulative net demand = 0.
- NMCS aircraft = sum of (2) and (3) = 68.
- Available aircraft = surviving acft - NMCS acft = 200 - 68 = 132.
- Achieved flying hours = the smaller of program hours for day 5 (1,500) and the product of available aircraft and maximum flying hours per aircraft per day (= 132 * 10 = 1,320) = 1,320.

(3) At this point the day convergence, C, is calculated as (difference in estimated versus achieved flying hours for this day)/(average daily program flying hours in war). Total program flying hours in war = 500 + 1,000 + 1,000 + 1,500 + 1,500 = 5,500. The daily average is 5500/5 = 1100. Therefore, C = (1,500 -1,320)/1,100 = .16. If C is less than the input convergence threshold (= 0 for this example), then iterations for the day stop. Otherwise, we check whether completed step (1) equals or exceeds the input-specified iteration limit (= 2 for this example). Since neither of these limits is satisfied, another iteration for day 5 proceeds as follows:

- New estimated flying hours = average of the estimated and achieved flying hours from last iteration = (1,500 + 1,320)/2 = 1,410.
- Part 1 cumulative net demand = cumulative failures from flying hours (500 + 1,000 + 1,000 + 1,500 + 1,410) - cumulative returned repairs -initial inventory = .08 * (5,410) - 120 - 252 = 61.
- Similarly, Part 2 net demand = 0.
- NMCS aircraft = 61 + 0 = 61.
- Available aircraft = 200 - 61 = 139.
- Achieved flying hours = min(1,500, 139*10) = 1,390.

(4) Convergence, C, for day = 5 * (1,410 - 1,390)/5,500 = .02. Since C equals or exceeds the convergence threshold, we check if this step (2) equals or exceeds our iteration limit. It does. Processing for this day is complete.

(5) Once all days are processed, final statistics are calculated based on the achieved flying hours from the last iteration of each day. The fraction flying hour program achieved is calculated by dividing the achieved flying hours for the day by the program flying hours for the day. Flying hours per (available) aircraft per day are calculated from achieved flying hours divided by available aircraft. Other calculations are analogous to those in Tables 2-20 and 2-21.

r. Capability Assessment of Constrained Cost Total Requirements. Stocking the total constrained cost solution (computed in Table 2-17) is equivalent, for this example, to stocking the residual (add-on) solution. Therefore, Tables 2-22 and 2-23 also apply to the total requirements solution.

Table 2-22. Assessment Iterations for Constrained Cost Residual Requirement

Day	Step	Estimated flying hours	Cumulative Net Demand Part 1 (init inv = 252)	Cumulative Net Demand Part 2 (init inv = 80)	NMCS acft	Available aircraft	Achieved flying hr
1	1	500	0	0	0	150	500
2	1	1,000	0	0	0	200	1,000
3	1	1,000	0	0	0	200	1,000
4	1	1,500	28	0	28	172	1,500
5	1	1,500	68	0	68	132	1,320
	2	1,410	61	0	**61**	**139**	**1,390**

Table 2-23. Capability Assessment for Constrained Cost Residual Requirement

Day	NMCS acft	Surviving acft	Aircraft availability	Fraction flying pgm acheived	Flying hour per acft per day
1	0	150	1.00	1.00	500/150 = 3.3
2	0	200	1.00	1.00	1,000/200 = 5.0
3	0	200	1.00	1.00	5.0
4	28	200	172/200 = .86	1.00	1,500/172 = 8.7
5	61	200	139/200 = .70	1,390/1,500 = .93	1,390/139 = 10.0

Avg availability = [(150 x 1.) + 2 x (200 x 1.) + (200 x .86) + (200 x .70)]/950 = .91

Avg frac pgm achieved = [(150 x 1.) + (200 x 1.) + (200 x 1.) + (172 x 1.) + (139 x .93)]/861 = .99

Avg flying hr/acft/day = [(150 x 3.3) + (200 x 5.0) + (200 x 5.0) + (172 x 8.7) + (139 x 10.0)]/861 = 6.3

```
C                                                OF PART. THUS IRC(J) IS THE 'PART NR' FOR
C                                                THE MOST EXPENSIVE PART TYPE.
C  QPA(J)               300  REAL               'QUANTITY PER APPLICATION' FOR PART J
C                                                I.E.  NR OF PART J  ITEMS INSTALLED PER AC
C
C  RNCS(J)              300  REAL               TOTAL REQMT(INIT STK=0) FOR PART J USING A
C                                                'NO SUBSTITUTION' REPLACEMENT POLICY
C                                                 WITH UNCONSTRAINED COST
C
C  RNCSI(J)             300  REAL               RESIDUAL REQMT(INIT STK=CURR STK) FOR PART J
C                                                USING A 'FULL SUBSTITUTION' REPLACEMENT POLICY
C                                                WITH UNCONSTRAINED COST
C
C  RNMCS(J)             300  REAL               TOTAL REQMT(INIT STK=0) FOR PART J USING A
C                                                'NMCS =0 ' REPLACEMENT POLICY WITH
C                                                UNCONSTRAINED COST
C
C  S(J)                 300  REAL               WORKING VARIABLE CONTAINING INITIALIZED STOCK
C                                                POSITION (EITHER INIT STK=0 OR INIT STK=
C                                                CURR STK) FOR PART J DURING PROCESSING IN
C                                                SUBROUTINES  NCRNCT AND NCRNCP
C
C
C
C  NOTEWORTHY SINGLE-SUBSCRIPT NAMES
C
C
C  NAME                      TYPE                         DESCRIPTION
C
C
C  ADDOST                    REAL               CONSTANT ADDED TO INPUT VALUE OF OST
C                                                (ORDER/SHIP TIME AS READ FROM OVERVIEW INPUT)
C                                                 TO YIELD THE OST USED IN PARCOM. THE OST
C                                                 IS THE SAME FOR ALL PART TYPES. ALSO
C                                                 OST=1-WAY TRAVEL TIME IN PARCOM.
C
C  AX                        REAL               AVERAGE DAILY MINIMUM REQUIRED ACFT AVAIL
C
C  BRR                       REAL               NR OF RETURNING REPAIRS ARRIVING FROM RETAIL
C                                                REPAIR ON A SPECIFIED DAY
C
C  CASE                      CHAR               CASE ID
C
C  CLNC                      REAL               COST LIMIT ON TOTAL BUY(INIT STK=0)
C                                                ASSUMING CURR INV IS 'REFUNDED' AND CASH
C                                                VALUE IS ADDED TO THE INPUT COST LIMIT(CLNCR)
C
C  CLNCR                     REAL               INPUT COST LIMIT (DOLLARS)ON ADD-ON BUY
C                                                (INIT STK=CURR INV)
C
C  CNC                       REAL               DIFFERENCE BETWEEN TOTAL COST OF UNCONSTR
C                                                COST 'NO SUB' SOLUTION AND CLNC
C
C  CONVF                     REAL               AN INPUT 'CONVERGENCE FACTOR'. FOR EACH DAY,
C                                                ITERATIONS TO ASSESS FLY HRS FLOWN(USING THE
C                                                CONSTR COST SOLUTION) CONTINUE UNTIL THE
C                                                ITERATION LIMIT (LIMIT) IS REACHED OR UNTIL THE
C                                                DISCREPANCY IN FLY HRS (INITIAL-FINAL) IS LESS
C                                                THAN (CONVF/NW)*TOTAL PGM FLY HRS FOR WAR
C
C  DRR                       REAL               NR OF RETURNING REPAIRS ARRIVING FROM DEPOT
C                                                REPAIR ON A SPECIFIC DAY
C
C  FHM                       REAL               MAXIMUM FLYING HRS PER ACFT PER DAY(INPUT)
C
C  IES                      FIXED               ESSENTIALITY CODE IN PART DATA INPUT. ONLY
C                                                PARTS WITH ESSENTIALITY CODE .LE. IESS ARE
C                                                PROCESSED
C
C  IESS                     FIXED               ITEM ESSENTIALITY CODE USED IN SELECTION OF
C                                                ESSENTIAL PARTS (SEE IES)
C
C  IMSEL                    FIXED               NUMBER OF PART TYPES FOR WHICH INDIV ITEM
C                                                'CUMULATIVE TOTAL RQMTS THRU DAY N' ARE
C                                                DESIRED (SEE SM(I,J),SNCM(I,J),SNM(I,J))
C
C  IOPT1                    FIXED               OPTION (0=OMIT,1=DO) TO PRINT REQUIREMENTS
C                                                LISTS FOR UNCONSTR COST TOTAL RQMTS SOLUTIONS
C
C  IOPT2                    FIXED               OPTION (0=OMIT,1=DO) TO PRINT REQUIREMENTS
C                                                LISTS FOR UNCONSTR COST RESIDUAL RQMTS SOL
C
```

```
C                                                 PART J (SR(I,J..)) AND ALLOWABLE BACKORDERS
C                                                 (ALLOWB(I)*QPA(J)) OVER DAYS 1,2,...
C
C                                                 LATER IN PROGRAM IT IS USED TO CALCULATE
C                                                 MINIMUM TOTAL STOCK REQMT (INIT STK=0)
C                                                  FOR PART J UNDER A 'NMCS=0' PARTS
C                                                 REPLACEMENT POLICY.    IT IS COMPUTED AS
C                                                    THE RUNNING MAXIMUM (OVER TIME) OF
C                                                  THE RUNNING MAXIMUM (OVER TIME) OF
C                                                  OVER DAYS 1,2,...
C
C SRMAX21(J)        300   REAL                    A WORKING VARIABLE INITIALLY USED TO CALC
C                                                 MINIMUM RESIDUAL STOCK REQMT (INIT STK=
C                                                 CURR STK) FOR PART J UNDER A 'FULL SUBST'
C                                                 REPLACEMENT POLICY.    IT IS COMPUTED
C                                                 AS THE RUNNING MAXIMUM (OVER TIME) OF
C                                                 THE DIFFERENCE BETWEEN NET DEMAND FOR
C                                                 PART J (SR(I,J..)) AND ALLOWABLE BACKORDERS
C                                                 (ALLOWB(I)*QPA(J)) OVER DAYS 1,2,...
C
C                                                 LATERIN THE PROGRAM IT IS USED TO CALC
C                                                 MINIMUM RESIDUAL STOCK REQMT (INIT STK=
C                                                 CURR STK) FOR PART J UNDER A 'NMCS=0'
C                                                 REPLACEMENT POLICY.    IT IS COMPUTED AS
C                                                 THE RUNNING MAXIMUM (OVER TIME) OF
C                                                 THE DIFFERENCE BETWEEN NET DEMAND IN
C                                                 EXCESS OF CURRENT STOCK  FOR PART J
C                                                 (SR(I,J..)-STK(J)) OVER DAYS 1,2,...
C
C STK(J)            300   REAL                    CURRENT STOCK OF PART J (=ISTK(J))
C
C SUMB(I)           120   REAL                    CUMULATIVE BACKORDERS (ALL PARTS ON DAY I
C
C ZLOSS(I)           61   REAL                    NUMBER OF DAILY AC LOSSES BY ATTRITION
C                                                 DURING I-TH TIME INTERVAL (IDAY(I) TO
C                                                 IDAY(I+1))
C
C ZNRT(J)           300   REAL                    NRTS (NOT REPAIRABLE THIS STATION) FRACTION
C                                                 FOR PART J
C
C
C
C COMMON BLOCK (UNLABELED) ENTRIES
C
C
C NAME      DIMENSION   TYPE                 DESCRIPTION
C
C
C ALLOWB(I)         120   REAL                    MAXIMUM ALLOWABLE NMCS AC ON DAY I WHICH
C                                                 WILL STILL ALLOW ACHIEVMENT OF CASE OBJECTIVE
C                                                 (FLY HR AND AVAILABILITY) ON DAY I
C
C BCY(J)            300   REAL                    BASE REPAIR CYCLE TIME (DAYS) OF PART J
C                                                 (=BASE REPAIR TIME OF PART)(BASE=RETAIL)
C
C BF(J)             300   REAL                    A COEFFICIENT USED IN CALCULATION OF NET
C                                                 DEMANDS(SR(I,J,...)) FOR PART J. IT=
C                                                 (1-BC(J))*(1-ZNRT(J))*CF(J)
C
C CF(J)             300   REAL                    A COEFFICIENT USED IN CALCULATION OF NET
C                                                 DEMANDS(SR(I,J,...)) FOR PART J. IT=
C                                                 FR(J)*QPA(J)
C
C DCY(J)            300   REAL                    DEPOT RECYCLE TIME FOR PART TYPE J,I.E.
C                                                 TIME BETWEEN REMOVAL AND RETURN FROM DEPOT
C                                                 REPAIR. THIS = DEPOT REPAIR TIME + 2*ORDER
C                                                 SHIP TIME.
C
C DF(J)             300   REAL                    A COEFFICIENT USED IN CALCULATION OF NET
C                                                 DEMANDS(SR(I,J,...)) FOR PART J. IT=
C                                                 (1-DC(J))*ZNRT(J)*CF(J)
C
C DMD(J)            300   REAL                    WORKING VARIABLE USED PRIMARILY IN CALC OF
C                                                 NET DEMAND(SR(I,J,..)) FOR PART J ON DAY I.
C                                                 WHEN (CUM)NET DMD THRU DAY I IS BEING
C                                                 CALCULATED,DMD(J) IS (CUM) NET DMD THRU THE
C                                                 PREVIOUS DAY.
C
C IPC(J)            300   FIXED                   ARRAY WHICH STORES THE PART NUMBER(BASED ON
C                                                 ORDER OF INPUT) CORRESPONDING TO THE J-TH
C                                                 RANK ORDER BASED ON DECREASING UNIT COST
C
```

```
C                                              CAPABILITY ASSESSMENT(IT IS ITERATED).
C
C  FHR(I)           120   REAL     FLYING HOUR REQUIREMENT FOR DAY I
C                                  BASED ON INPUT FLYING HR PROGRAM
C
C  FR(J)            300   REAL     FAILURE(REPLACEMENT) RATE FOR PART J
C                                  EXPRESSED AS EXPECTED NR OF FAILURES
C                                  PER FLYING HOUR FLOWN.
C
C  IDAY(I)           61   FIXED    ARRAY WHICH TEMPORARILY STORES INPUT DATA ON
C                                  DAYS BEGINNING 'DAY INTERVALS' (IDAY(I) TO
C                                  IDAY(I+1)) IN WHICH VARIOUS INPUT DATA
C                                  TAKE EFFECT
C
C  INC(J)           300   FIXED    TOTAL 'SOLUTION STOCK'(CURR INV + ADD-ON REQ)
C                                  FOR PART TYPE J FOR UNCONSTR COST CASE
C                                  (RESIDUAL REQUIREMENTS ONLY)
C
C  IPT(J)             5   FIXED    ARRAY STORING INTERNAL PART NRS (SUBSCRIPTS)
C                                  FOR PARTS FOR WHICH A CUMULATIVE DAY BY DAY
C                                  REQUIREMENT HISTORY IS TO BE  PRINTED
C                                  (FOR INIT STK=0 ONLY)
C
C  ISTK(J)          300   FIXED    CURRENT STOCK OF PART J (INPUT)
C
C  NAC(I)            61   FIXED    NR OF AC DEPLOYED AT START OF I-TH TIME
C                                  INTERVAL (IDAY(I) TO IDAY(I+1))
C
C  NFH(I)            61   FIXED    FLYING HR REQMT DURING I-TH TIME
C                                  INTERVAL (IDAY(I) TO IDAY(I+1))
C
C  QPA(J)           300   REAL     'QUANTITY PER APPLICATION' FOR PART J
C                                  I.E.  NR OF PART J  ITEMS INSTALLED PER AC
C
C  RAV(I)           120   CHAR     DESIGNATES DOMINANT SOURCE OF 'MINIMUM
C                                  REQUIRED ACFT AVAIL' FOR DAY I,EITHER
C                                  'FLYING HR PROG' OR 'AVAIL CONSTR',I.E.
C                                  EITHER INPUT FLYING HR PROGRAM GOAL OR
C                                  ACFT AVAILABILITY CONSTRAINT(BOTH INPUT)
C
C
C  RFC(I)           120   REAL     AC AVAILABILITY WHEN TOTAL REQ(INIT STK=0)
C                                  IS STOCKED USING A 'FULL SUBSTITUTION'
C                                  REPLACEMENT POLICY WITH UNCONSTRAINED COST
C
C  RFCS(J)          300   REAL     TOTAL REQMT(INIT STK=0) FOR PART J USING A
C                                  'FULL SUBSTITUTION' REPLACEMENT POLICY
C                                   WITH UNCONSTRAINED COST
C
C  RFCS1(J)         300   REAL     RESIDUAL REQMT(INIT STK=CURRSTK) FOR PART J
C                                  USING A 'FULL SUBSTITUTION' REPLACEMENT POLICY
C                                   WITH UNCONSTRAINED COST
C
C  RNC(I)           120   REAL     AC AVAILABILITY WHEN TOTAL REQ(INIT STK=0)
C                                  IS STOCKED USING A 'NO SUBSTITUTION'
C                                  REPLACEMENT POLICY WITH UNCONSTRAINED COST
C
C  RNMCS1(J)        300   REAL     RESIDUAL REQMT(INIT STK=CURR STK) FOR PART J
C                                  USING A 'NMCS =0' REPLACEMENT POLICY
C                                  WITH UNCONSTRAINED COST
C
C  SM(I,J)        120,5   REAL     CUMULATIVE TOTAL (INIT STK=0) REQMT FOR
C                                  PART IPT(J) THRU DAY I WITH A FULL SUBSTITUTION
C                                  REPLACEMENT POLICY WITH UNCONSTRAINED COST
C                                  (SEE DESCRIPTION OF IPT(J))
C
C  SNCM(I,J)      120,5   REAL     CUMULATIVE TOTAL (INIT STK=0) REQMT FOR
C                                  PART IPT(J) THRU DAY I WITH A NO SUBSTITUTION
C                                  REPLACEMENT POLICY WITH UNCONSTRAINED COST
C                                  (SEE DESCRIPTION OF IPT(J))
C
C  SNM(I,J)       120,5   REAL     CUMULATIVE TOTAL (INIT STK=0) REQMT FOR
C                                  PART IPT(J) THRU DAY I WITH A 'NMCS=0'
C                                  REPLACEMENT POLICY WITH UNCONSTRAINED COST
C                                  (SEE DESCRIPTION OF IPT(J))
C
C  SRMAX1(J)        300   REAL     A WORKING VARIABLE USED INITIALLY TO CALC
C                                  MINIMUM TOTAL STOCK REQMT (INIT STK=0)
C                                  FOR PART J UNDER A 'FULL SUBSTITUTION'
C                                  REPLACEMENT POLICY.    IT IS COMPUTED
C                                  AS THE RUNNING MAXIMUM (OVER TIME) OF
C                                  THE DIFFERENCE BETWEEN NET DEMAND FOR
```

```
C
C CNCS(J)          300   REAL      TOTAL COST (INIT STK=0) OF REQMT FOR PART J
C                                 USING A   NO SUSTITUTION  REPLACEMENT POLICY
C                                 WITH UNCONSTR COST WITH CONSTR COST
C
C CNCSI(J)         300   REAL      RESIDUAL COST (INIT STK=CURR STK)OF REQMT
C                                 (IN EXCESS OF CURRENT STOCK) FOR
C                                 PART J USING A   NO SUSTITUTION  REPLACEMENT
C                                 POLICY WITH UNCONSTR COST WITH CONSTR COST
C
C CNMCS(J)         300   REAL      TOTAL COST (INIT STK=0) OF REQMT FOR PART J
C                                 USING A   'NMCS=0'     REPLACEMENT POLICY
C                                 WITH UNCONSTRAINED COST
C
C CNMCSI(J)        300   REAL      RESIDUAL COST (INIT STK=CURR STK)OF REQMT
C                                 (IN EXCESS OF CURRENT STOCK) FOR
C                                 PART J USING A   'NMCS=0'   REPLACEMENT
C                                 POLICY WITH UNCONSTRAINED COST
C
C COST(J)          300   REAL      COST OF A SINGLE ITEM OF PART J
C
C DC(J)            300   REAL      DEPOT CONDEMNATION RATE OF PART J
C                                 (FRAC FAILURES 'JUNKED' AT DEPOT LEVEL)
C
C DMDT(J)          300   REAL      TEMPORARY VALUE FOR DMD(J) USED WHILE
C                                 CONVERGING TO A SINGLE'FLYING HRS FLOWN'
C                                 RESULT DURING CAPABILITY ASSESSMENT
C
C DCOSTF(I)        120   REAL      CUMULATIVE COST OF TOTAL (INIT STK=0) REQMT
C                                 (ALL  PARTS) THRU DAY I USING A FULL
C                                 SUBSTITUTION POLICY WITH UNCONSTRAINED COST
C
C DCOSTN(I)        120   REAL      CUMULATIVE COST OF TOTAL (INIT STK=0) REQMT
C                                 (ALL  PARTS) THRU DAY I USING A NO
C                                 SUBSTITUTION POLICY WITH UNCONSTRAINED COST
C
C DCOSTZ(I)        120   REAL      CUMULATIVE COST OF TOTAL (INIT STK=0) REQMT
C                                 (ALL  PARTS) THRU DAY I USING A 'NMCS=0'
C                                 REPLACEMENT  POLICY WITH UNCONSTRAINED COST
C
C ECOSTF(I)        120   REAL      CUMUL COST OF RESIDUAL (INIT STK=CURR STK)
C                                 REQMT (IN EXCESS OF CURRENT STOCK) THROUGH
C                                 DAY I USING A FULL SUBSTITUTION REPLACEMENT
C                                 POLICY WITH UNCONSTRAINED COST
C
C ECOSTN(I)        120   REAL      CUMUL COST OF RESIDUAL (INIT STK=CURR STK)
C                                 REQMT (IN EXCESS OF CURRENT STOCK) THROUGH
C                                 DAY I USING A   NO SUBSTITUTION REPLACEMENT
C                                 POLICY WITH UNCONSTRAINED COST
C
C ECOSTZ(I)        120   REAL      CUMUL COST OF RESIDUAL (INIT STK=CURR STK)
C                                 REQMT (IN EXCESS OF CURRENT STOCK) THROUGH
C                                 DAY I USING A   'NMCS=0'   REPLACEMENT POLICY
C                                 WITH UNCONSTRAINED COST
C
C FHA(I)           120   REAL      INITIAL ESTIMATE FOR FLYING HRS ACHIEVED ON
C                                 DAY I WHEN COMPUTED REQMT - BASED ON A COST
C                                 CONSTRAINED NO SUBSTITUTION REPLACEMENT POLICY-
C                                 IS STOCKED. (COMPUTED RECURSIVELY)
C
C FHPAPD(K,I)    3,120   REAL      FHPAPD(1,I)=FLYING HRS PER AVAILABLE ACFT PER
C                                 FOR DAY I UNDER FULL SUB USING THE UNCONSTR
C                                 COST SOLUTION STOCK.
C
C                                 FHPAPD(2,I)=FLYING HRS PER AVAILABLE ACFT PER
C                                 FOR DAY I UNDER NO SUB USING THE UNCONSTR
C                                 COST SOLUTION STOCK.
C
C                                 FHPAPD(3,I)=FLYING HRS PER AVAILABLE ACFT PER
C                                 FOR DAY I UNDER NO SUB USING THE CONSTRAINED
C                                 COST SOLUTION STOCK.
C
C FHNC(I)          120   REAL      CALCULATED ESTIMATE FOR FLYING HRS ACHIEVED ON
C                                 DAY I WHEN COMPUTED REQMT - BASED ON A COST
C                                 CONSTRAINED NO SUBSTITUTION REPLACEMENT POLICY-
C                                 IS STOCKED. (COMPUTED RECURSIVELY)
C
C FHNZ(I)          120   REAL      FRACTION OF FLYING PROGRAM COMPLETED ON DAY I
C                                 WHEN COST-CONSTRAINED SOLUTION IS STOCKED
C                                 USING A 'NO SUBSTITUTION' REPLACEMENT POLICY
C
C FHA(I)           120   REAL      ESTIMATED FLY HRS FLOWN ON DAY I DURING
```

MAIN PROGRAM

```
C NAME: PARCOM            TYPE: MAIN PROGRAM
C
C PURPOSE: THE PARCOM (PARTS REQUIREMENTS AND COST MODEL) IS USED TO GENERATE
C COST EFFECTIVE MIXES OF SPARE PARTS NEEDED TO ACHIEVE A FLYING PROGRAM UNDER
C VARIOUS
C     -PART REPLACEMENT POLICIES
C     -NMCS (NOT MISSION CAPABLE SUPPLY) CONSTRAINTS
C     -PART INVENTORY COST CONSTRAINTS
C
C ARGUMENTS: NOT  APPLICABLE
C
C CALLED BY: NOT  APPLICABLE
C
C CALLS
C     -FUNCTION MAXC
C     -FUNCTION SR
C     -SUBROUTINE NCRNCT
C     -SUBROUTINE NCRNCR
C
C FILES USED : INPUT - UNIT 10 (PARTS DATA)
C                    - UNIT 11 (SCENARIO DATA)
C             OUTPUT - PRINT
C             TEMP   - UNIT 12 -FILE WITH (CURR INV + ADD-ON RQMNT) FOR
C                               EACH PART TYPE(W/ NONZERO FAILURE RATE)
C                               IN ORDER OF INPUT FOR THIS RUN,BASED ON
C                               UNCONSTR COST.  FILE IS IN FORMAT FOR
C                               PARCOM INPUT(ISTK(J)).
C
C LOCAL ARRAYS
C
C
C NAME        DIMENSION   TYPE              DESCRIPTION
C
C
C AC(I)             120   REAL      NR  ACFT DEPLOYED ON DAY I
C
C ADESC(J)          300   CHAR      16 CHAR DESCRIPTION OF SPARE PART J
C
C ALR(I)            120   REAL      NR ACFT LOST(ATTRITION) ON DAY I
C
C AM(I)              61   REAL      AC AVAILABILITY CONSTRAINT FOR I-TH
C                                   'DAY INTERVAL',I.E. MINIMUM REQUIRED ACFT
C                                   AVAILABILITY IN I-TH 'DAY INTERVAL'
C
C AMSN(J)           300   CHAR      IDENTIFICATION NR(NSN) OF SPARE PART J
C
C ASURV(I)          120   REAL      NR AC SURVIVING (NOT ATTRITTED)ON DAY I
C
C AVAVG(K)            6   REAL      INITIALLY AVAVG(1)=AVG ACFT AVAIL W/FULL
C                                   SUB SOLUTION-UNCONSTR COST. LATER AVAVG(1)=
C                                   AVG ACFT AVAIL W/NO SUB SOL.-CONSTR COST.
C                                   AVAVG(2)=AVG ACFT AVAIL W/NO SUB SOL.-
C                                   UNCONSTR COST.
C
C                                   AVAVG(3)=AVG MIN ACFT REQ'D TO ACHIEVE
C                                   GOAL/OBJECTIVE(ALL COST CONDITIONS)
C
C                                   AVAVG(4)=AVG FLY HR/AVAIL ACFT / DAY
C                                   W/FULL SUB SOL.-UNCONSTR COST
C
C                                   AVAVG(5)=AVG FLY HR/AVAIL ACFT / DAY
C                                   W/NO SUB SOL.-UNCONSTR COST.
C
C                                   AVAVG(6)=AVG FLY HR/AVAIL ACFT / DAY
C                                   W/NO SUB SOL.-CONSTR COST.
C
C AVM(I)            120   REAL      AC AVAILABILITY CONSTRAINT FOR DAY I
C
C BC(J)             300   REAL      BASE(RETAIL) CONDEMNATION RATE OF PART J
C                                   (=FRACTION FAILURES 'JUNKED' AT RETAIL LEVEL)
C
C CFCS(J)           300   REAL      TOTAL COST (INIT STK=0) OF REQMT FOR PART J
C                                   USING A FULL SUSTITUTION  REPLACEMENT POLICY
C                                   WITH UNCONSTRAINED COST
C
C CFCS1(J)          300   REAL      RESIDUAL COST (INIT STK=CURR STK)OF REQMT
C                                   (IN EXCESS OF CURRENT STOCK) FOR
C                                   PART J USING A FULL SUSTITUTION  REPLACEMENT
C                                   POLICY WITH UNCONSTRAINED COST
C
C CLOSS(I)          120   REAL      CUMULATIVE AC  LOST TO ATTRITION THRU DAY I
```

(NOT USED)

APPENDIX A

PARCOM PROGRAM SOURCE CODE

4-5. CAVEATS. If the day and/or parts limits are increased, the processing time required for a PARCOM execution increases by at least the product of the two limit multipliers, i.e., doubling the day limit and the part limit will at least quadruple processing time. In the PARCOM User's Guide, it was noted that the calculation of entries for "no substitution" in the output "Cumulative Total Requirement Cost List" and the output "Cumulative Residual Requirement Cost List" consumes most of the processing time required for a PARCOM run. Therefore, if those entries are not essential, time-consuming calculations can be minimized by setting IPRT4 = the day limit in record set 4 of the Scenario Input Data Base as described in the PARCOM User's Guide.

Table 4-1. PARCOM Arrays with a Day Limit Dimension

Array	Routine	Array	Routine	Array	Routine
AC(120)	Main	ECOSTZ(120)	Main	SNM(120,5)	Main
ALR(120)	Main	FHNC(120)	Main	SUMB(120)	Main
ASURV(120)	Main	FHNZ(120)	Main	FHR(120)	NCRNCT
AVM(120)	Main	FHAPD(3,120)	Main	FHR(120)	NCRNCR
ALLOWB(120)	COMMON	FHR(120)	Main	SUMBZ(120)	NCRNCT
DCOSTF(120)	Main	IFCD(120)	Main	SUMBZ(120)	NCRNCR
DCOSTN(120)	Main	RFC(120)	Main	SUMP(120)	NCRNCR
DCOSTZ(120)	Main	RNC(120)	Main	FHR(120)	SR
ECOSTF(120)	Main	SM(120,5)	Main	FHA(120)	Main
ECOSTN(120)	Main	SNCM(120,5)	Main	RAV(120)	Main

4-4. EXTENSION OF TOTAL PARTS LIMIT. In the PARCOM version delivered by CAA, 34 single-subscript arrays are defined in terms of the maximum number of parts to be processed. The current limit is 300 parts. Those arrays of size 300 may be increased in size (through user reprograming) to any limit permitted by computer memory. The arrays associated with the parts limit and the routines defining them are shown in Table 4-2.

Table 4-2. PARCOM Arrays with a Parts Limit Dimension

Array	Routine	Array	Routine	Array	Routine
BC(300)	Main	INC(300)	Main	BF(300)	COMMON
CFCS(300)	Main	ISTK(300)	Main	CF(300)	COMMON
CFCS1(300)	Main	QPA(300)	Main	DCY(300)	COMMON
CNCS(300)	Main	RFCS(300)	Main	DF(300)	COMMON
CNCS1(300)	Main	RFCS1(300)	Main	DMD(300)	COMMON
CNMCS(300)	Main	RNMCS1(300)	Main	IRC(300)	COMMON
CNMCS1(300)	Main	SRMAX1(300)	Main	RNCS(300)	COMMON
COST(300)	Main	SRMAX2(300)	Main	RNCS1(300)	COMMON
DC(300)	Main	STK(300)	Main	RNMCS(300)	COMMON
DMDT(300)	Main	ZNRT(300)	Main	S(300)	COMMON
FR(300)	Main	BCY(300)	COMMON	ADESC(300)	Main
				AMSN(300)	Main

(5) Assesses the combat capability of the unconstrained cost "full substitution" and "no substitution" solution mixes.

(6) Computes all requirements for the constrained cost cases.

(7) Assesses the combat capability of the constrained cost solution mixes.

b. Function MAXC. Function MAXC is called only by the main program. No external routines are called by it. Function MAXC determines the subscript of the largest (in value) member of an array. Repeated application enables ordering of an array. In this way, the main program orders all part types in order of decreasing part unit cost. Function MAXC is also used to determine the smallest of the input "minimum required aircraft availability" specifications (for use in an output header).

c. Subroutine NCRNCT. Subroutine NCRNCT is called by the main program and calls function SR. This subroutine computes a least cost total requirements mix (zero initial inventory) for the unconstrained cost case with a "no substitution" policy.

d. Subroutine NCRNCR. Subroutine NCRNCR is called by the main program and calls function SR. This subroutine computes a least cost residual requirements mix (add-on to specified initial inventory) for the unconstrained cost case with a "no substitution" policy.

e. Function SR. Function SR is called by the main program, by subroutine NCRNCT, and by subroutine NCRNCR. It calculates the cumulative net demand through a specified day for a specified part (based on a specified flying program) prior to the adjustment (subtraction) for initial inventory.

4-2. ARRAY STORAGE. Definitions and sizes of PARCOM array variables are given in the comments of the program code displayed in Appendix A. The types of arrays are local, as defined by DIMENSION statements, common, as defined by unlabeled COMMON, and character, as defined by CHARACTER declarations. Character variables occupy four words per entry in PARCOM while other arrays require only one word per entry. During execution on the Sperry 1100/82 computer, PARCOM occupies 32,768 words of memory. Of this, 18,076 words, or 53 percent of total requirements are associated with arrays.

4-3. EXTENSION OF DAY LIMIT. In the PARCOM version delivered by CAA, 26 single-subscript arrays and 4 double-subscript arrays are defined in terms of the maximum number of days in the scenario. The current limit is 120 days. Those arrays of size 120 may be increased in size (through user reprograming) to the scenario length desired insofar as computer memory permits. The arrays associated with the day limit, their dimensions, and the routines defining them are listed in Table 4-1.

CHAPTER 4

POTENTIAL PROGRAM MODIFICATION

4-1. MODULE FUNCTIONS. Figure 4-1 shows the main and subprogram modules of PARCOM. The subprograms consist of two subroutines and two functions. A summary of operational purpose is given below for each module. Details of module operations can be read in the commented FORTRAN code for PARCOM presented in Appendix A.

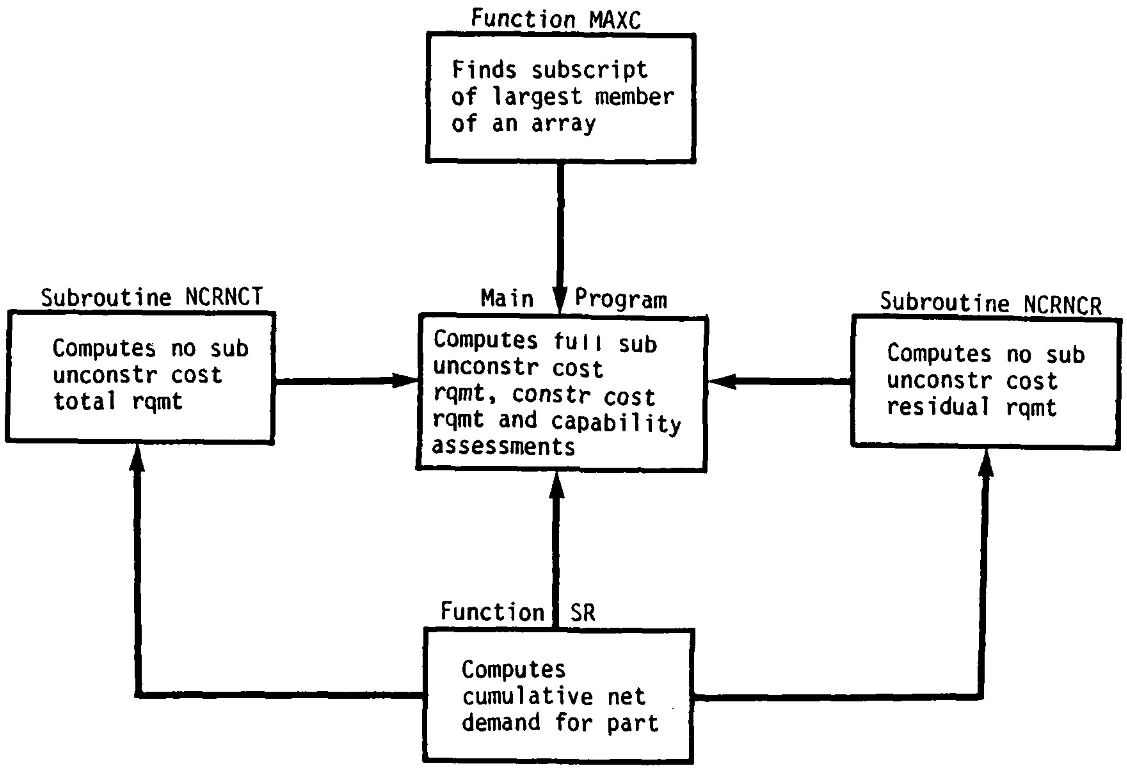


Figure 4-1. PARCOM Subprogram Modules

a. **Main Program.** The PARCOM main program:

(1) Reads in all part and scenario data.

(2) Computes all requirements and costs for unconstrained cost cases with full substitution.

(3) Computes all requirements and costs for unconstrained cost cases with a "NMCS = 0" policy.

(4) Calls subroutines NCRNCT and NCRNCR to obtain requirements and costs for unconstrained cost cases with a "no substitution" policy. Calls the functions MAXC and SR for reasons discussed later.

g. **No Stochastic Results.** All PARCOM results are "expected value." Neither input nor results have variable probabilistic aspects (e.g., confidence levels). Safety levels would have to be treated separately as an add-on to PARCOM quantities. However, use of expected values is meaningful for comparisons and parametric evaluations. Methodology for incorporating stochastic considerations into PARCOM would be complex. Conversion of the model into a stochastic simulation could entail high risk for an uncertain payoff.

availabilities are determined. At times, however, it is also desirable to be able to assess the degree to which an aircraft fleet, with its current or some other starting inventory (and no add-ons), can meet specified flying or availability goals. This can be done for the "no substitution" policy by running that inventory in a requirements mode with a constrained cost limit of zero (so that no parts will be added). In that case, the full complement of assessment outputs is available. With "full substitution", however, only the expected period (consecutive days from day 1) of full flying hour program sustainability is assessable for current inventory. This is accomplished by running an unconstrained cost "full substituton" requirements case and reading from the cumulative cost output the last day that no additional costs are experienced.

3-3. CAVEATS AND LIMITATIONS. The principal caveats and limitations on the PARCOM Model, as applied in the study, are discussed below. Program modification and/or restructuring is required to extend model capabilities beyond the cited limits.

a. Number of Part Types Processed. The PARCOM Model version demonstrated herein can process at most 300 different part types. Simple (but memory consuming) modifications to the structure of the program can significantly increase this capacity.

b. No "Partial Substitution". PARCOM currently processes only "full substitution," "no substitution," and "NMCS = 0" policies. There is no definitive logic yet for a "partial substitution" policy. In light of underlying data and process uncertainties, the bounds of costs and amounts reflected in the "no substitution" and "full substitution" solutions may well be sufficient.

c. Incomplete Assessment of "Full Substitution" Constrained Cost Solutions. Additional programing effort might enable a complete assessment of "full substitution" constrained cost solutions. Currently only the days of sustainability can be determined.

d. Only Two Centralized Supply Levels. PARCOM shares the Overview Model "world view" of a retail level and a wholesale level. With "full substitution", each level has full cross-leveling (lateral transferability) of parts.

e. No Indenture Levels. Part types in the PARCOM (and Overview) data base are nonoverlapping modular units, i.e., no part is a subcomponent of another listed part type. Therefore, the failures and repair of parts are independent of each other. Use of indentured data is not processable in PARCOM.

f. No Direct Maintenance Modeling. As with Overview, PARCOM treats maintenance only indirectly, by incorporation in the repair time or by using an aircraft deployment/attrition data base which is adjusted for aircraft down ("lost") due to maintenance constraints. Such adjustments could be based on results of a separate high-resolution simulation model which previously processed a "slice" of the scenario.

CHAPTER 3

OPERATIONAL CONSIDERATIONS AND CAVEATS

3-1. CASE OBJECTIVES. The user can specify a flying hour objective in conjunction with an aircraft availability objective. For each of these, one of two subobjectives is selected. The associated case types are noted below.

a. Maximizing Cumulative Flying Hours Achieved. This flying hour objective is always operating when running a constrained cost case. It entails the direct determination of the parts mix which will yield the greatest number of achieved flying hours for a specified cost limit. The flying hours achieved will be less than the desired flying hour program if the cost limit is less than the cost of the unconstrained cost solution mix.

b. Maximizing Consecutive Daily Program Flying Hours Achieved. This flying hour objective is relevant only to constrained cost cases since, for unconstrained cost cases, achieved flying hours = program flying hours. Obtaining a solution with this objective is a two-stage process. First, the user runs PARCOM in an unconstrained cost mode for the full wartime period. The output list from that run shows, for each day, the cumulative cost of the add-on parts that would have been required if the war had been truncated at that day. D, the last day on that list for which the associated cost is less than or equal to the "cost limit" of the constrained cost case, is then the maximum number of consecutive days of 100 percent flying program sustainability with "cost limit" spares dollars. Next, to get the solution mix associated with D, PARCOM is rerun, in the unconstrained cost mode, with a truncated "war" of D days length.

c. Minimum Specified Daily Aircraft Availability. This objective is in addition to any flying hour objective and is operative in all cases. The availability objective may increase the demand for available aircraft beyond those required to achieve the flying program. The input availability constraints are, as described previously, used to calculate daily "allowed NMCS aircraft", which, in turn, is used in all case calculations.

d. No Specified Aircraft Availability. PARCOM must always read in values for minimum daily aircraft availability objectives. However, entering blank or zero equates to not specifying an availability objective.

3-2. CAPABILITY ASSESSMENT. Normally, PARCOM capability assessments are performed when add-on requirements are determined for both unconstrained and constrained cost cases. In the unconstrained cost cases, flying hour and availability goals are fully met, so the assessed achievements are simply the same as the goals. However, average availability over the course of the war, which cannot be input as a goal, is also determined. For constrained cost cases days of sustainability, fraction of daily and total flying hour program achieved, and daily and average aircraft

```
      C IOPT3                 FIXED    OPTION (0=OMIT,1=DO) TO PRINT REQUIREMENTS
      C                                LISTS FOR CONSTR COST 'TOTAL BUY' SOLUTION
      C                                (BASED ON   COST LIMIT= VALUE OF CURR INV+
      C                                INPUT COST LIMIT APPLIED TO INIT STK=0)
      C
      C IOPT4                 FIXED    OPTION (0=OMIT,1=DO) TO PRINT REQUIREMENTS
      C                                LISTS FOR CONSTR COST 'ADD-ON BUY' SOLUTION
      C                                (BASED ON INPUT COST LIMIT APPLIED TO
      C                                INIT STK=CURR INV)
      C
      C IOPT5                 FIXED    OPTION (0=OMIT,1=DO) TO PRINT TOTAL
      C                                (INIT STK =0)  CUMULATIVE (BY DAY) RQMNTS
      C                                FOR SELECTED ITEMS (SEE IMSEL,SM(I,J),
      C                                SNCM(I,J),SNM(I,J))
      C
      C IOPT6                 FIXED    OPTION (0=OMIT,1=DO) TO DO CONSTR COST CASE
      C
      C IPRT                  FIXED    OPTION (0=OMIT,1=DO) TO PRINT SCENARIO
      C                                DATA BASE SUMMARY OUTPUT LIST
      C
      C
      C IPRT4                 FIXED    INTERVAL(DAYS) AT WHICH 'CUMULATIVE RQMNTS
      C                                COST THRU DAY N' ARE CALCULATED FOR 'NO SUB'
      C                                IN THE UNCONSTRAINED COST CASE
      C
      C KNT                   FIXED    INDICATOR 1=TOTAL RQMNTS BEING PROCESSED,
      C                                2=RESIDUAL(ADD-ON) RQMNTS BEING PROCESSED
      C
      C LIMIT                 FIXED    MAXIMUM NR OF ASSESSMENT ITERATIONS(EACH
      C                                DAY) SEEKING CONVERGENCE OF FLY HRS FLOWN
      C
      C NP                    FIXED    NR OF PART TYPES PROCESSED IN RUN. (THIS
      C                                EXCLUDES PART TYPES WITH ESSENTIALITY CODE
      C                                .LE. IESS OR WITH A ZERO FAILURE RATE)
      C
      C NW                    FIXED    LENGTH(DAYS) OF SCENARIO
      C
      C TCFCS                  REAL    TOTAL(PROCUREMENT) COST OF 'UNCONSTR COST
      C                                TOTAL(INIT STK=0) RQMNT' SOLUTION WITH
      C                                FULL SUBSTITUTION
      C
      C TCNCS                  REAL    TOTAL(PROCUREMENT) COST OF 'UNCONSTR COST
      C                                TOTAL(INIT STK=0) RQMNT' SOLUTION WITH
      C                                NO SUBSTITUTION
      C
      C TCNMCS                 REAL    TOTAL(PROCUREMENT) COST OF 'UNCONSTR COST
      C                                TOTAL(INIT STK=0) RQMNT' SOLUTION WITH
      C                                'NMCS=0'
      C
      C UCFCS                  REAL    TOTAL(PROCUREMENT) COST OF 'UNCONSTR COST
      C                                RESIDUAL(ADD-ON) RQMNT' SOLUTION WITH
      C                                FULL SUBSTITUTION
      C
      C UCNCS                  REAL    TOTAL(PROCUREMENT) COST OF 'UNCONSTR COST
      C                                RESIDUAL(ADD-ON) RQMNT' SOLUTION WITH
      C                                NO SUBSTITUTION
      C
      C UCNMCS                 REAL    TOTAL(PROCUREMENT) COST OF 'UNCONSTR COST
      C                                RESIDUAL(ADD-ON) RQMNT' SOLUTION WITH
      C                                'NMCS=0'
      C
            DIMENSION
           +    AC(120),        ALR(120),       AM(61),         ASURV(120),
           +    AVAVG(6),       AVM(120),       BC(300),        CFCS(300),
           +    CFCS1(300),                     CNCS(300),      CNCS1(300),
           +    CNMCS(300),     CNMCS1(300),    COST(300),      DC(300),
           +    DCOSTF(120),    DCOSTN(120),    DCOSTZ(120),    DMDT(300),
           +    ECOSTF(120),    ECOSTN(120),    ECOSTZ(120),    FHA(120),
           +    FHNC(120),      FHNZ(120),      FHPAPD(3,120),  FHR(120),
           +    FR(300),        IDAY(61),                       INC(300),
           +    IPT(5),         ISTK(300),      NAC(61),        NFH(61),
           +    QPA(300),       RFC(120),       RFCS(300),      RFCS1(300),
           +    RNC(120),                       RNMCS1(300),    SM(120,5),
           +    SNCM(120,5),    SNM(120,5),     SRMAX1(300),    SRMAX2(300),
           +    STK(300),       SUMB(120),      ZLOSS(61),      ZNPT(300)
            EQUIVALENCE
           + (QPA,DMDT)   ,    (ISTK,INC), (SRMAX1,CFCS), (SRMAX2,CNCS),
           + (DC,CNMCS)   ,    (BC,RFCS1), (DCOSTF,FHA ), (DCOSTN,FHNC),
           + (DCOSTZ,FHNZ),    (ALR,RFC )
            COMMON
           +    ALLOWB(120),    BCY(300),       BF(300),        CF(300),
           +    DCY(300),       DF(300),        DMO(300),       IRC(300),
```

```
     +    RNCS(300),       RNCSI(300),     RNMCS(300),     S(300)
      CHARACTER*16
     +    ADESC(300),      ADSC,           AMSN(300),      CASE,
     +    RAV(120),        Z1
      DO 100 I=1,61
        IDAY(I)=0
        AVM(I)=0.
        NAC(I)=0
        NFH(I)=0
        ZLOSS(I)=0
  100 AM(I)=0.
      ZZ=0.
C
C
C  READ (CARD IMAGE)ORDER SHIP TIME OFFSET,FLY HR CONVERGENCE FACTOR
C  AND PART ESSENTIALITY THRESHOLD
C
C
      READ (11,9700) ADDOST,CONVF,IESS
      NP=0
C
C  THIS SECTION READS(102 CHAR RECORDS ON LOGICAL UNIT 11) THE PARTS
C  DATA BASE. DATA FOR EACH PART ARE CONTAINED IN SETS OF 12 CONSECUTIVE
C  RECORDS(ONLY 3 OF WHICH ARE READ). INITAIALLY SKIP 3 RECORDS WHICH
C  HEAD THE FILE BUT ARE NOT PART OF ANY 'PART DATA SET'.
C
      READ (10,9800)
      I=0
  200 READ (10,9900,END=600) Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,Z9,IES,INIT
C
C               SKIP 3 RECORDS
C
      READ (10,9800,END=600)
C
C               READ QUANTITY PER APPLICATION
C
      READ (10,10000,END=600) IQPA
C
C  READ PART DESCRIPTION,THEN SKIP 6 RECORDS.THEN SCREEN TO PROCESS ONLY
C  PARTS WITHIN ESSENTIALITY THRESHOLD
C
      READ (10,10100,END=600) ADSC
      IF (IES.GT.IESS) GO TO 400
C
C               COMPUTE TOTAL DEPOT CYCLE TIME.IN DOING SO ADJUST INPUT
C               OST BY ADDOST
C
      ZXD=2.*Z3+Z7+2.*ADDOST
      Z2C=Z2/100.
      Z4F=Z4/1000000.
      Z5N=Z5/100.
      Z100=IQPA
      Z8B=Z8/100.
      Z9D=Z9/100.
      IF (MOD(NP+I,50).NE.0) GO TO 300
      WRITE (6,10200)
      WRITE (6,10300)
      WRITE (6,10400)
      WRITE (6,10500)
C
C  SCREEN TO PROCESS ONLY PARTS WITH NONZERO FAILURE RATES. COUNT ALL
C  PARTS(INDEX=I). ALWAYS  PRINT A PART DATA  RECORD FOR EVERY INPUT PART,
C  BUT CHECK IF PART HAS A  NON-ZERO FAILURE RATE OR IS 'NONESSENTIAL'.
C  IF SO,OMIT 'PART NR'(NP),OTHERWISE PRINT THE PART NR TOO.
C
  300 IF (Z4.GE..0000001) GO TO 500
  400 WRITE (6,10600) Z1,ADSC,Z2C,Z3,Z4F,Z5N,Z6,ZXD,Z7,Z8B,Z9D,Z100,IES
      I=I+1
      GO TO 200
  500 NP=NP+1
      WRITE (6,10700) NP,Z1,ADSC,Z2C,Z3,Z4F,Z5N,Z6,ZXD,Z7,Z8B,Z9D,Z100,I
     +ES,INIT
      AMSN(NP)=Z1
      COST(NP)=Z2C
      FR(NP)=Z4F
      ZNRT(NP)=Z5N
      BCY(NP)=Z6
      DCY(NP)=ZXD
      ISTK(NP)=INIT
      BC(NP)=Z8B
      DC(NP)=Z9D
      QPA(NP)=Z100
```

```
      ADESC(NP)=ADSC
      GO TO 200
  600 II=NP+I
C
C  PRINT THE TOTAL NR OF PARTS INPUT (NP+I) AND THE TOTAL NR  OF
C  PARTS TO BE PROCESSED(NP).
C
      WRITE (6,10800) II,NP
C
C READ (CARD IMAGE) THE REST OF THE INPUT FACTORS
C    CARD 2- CASE ID
C    CARD 3= COST LIMIT,ITERATION LIMIT
C    CARD 4=MAX FLY HR/AC/DAY,NR DAYS IN WAR,OUTPUT PRINT OPTIONS
C FOR A VARIETY OF INPUTS TO FOLLOW,READ,IN SEQUENCE
C      -NR OF 'DAY INTERVALS' SPECIFIE
C      -INITIAL DAY OF EACH 'DAY INTERVAL'
C      -THE STATUS(OF INPUT FACTOR) AT START OF EACH 'DAY INTERVAL'
C THEREAFTER,SET STATUS VALUE FOR EACH DAY OF INTERVAL(=INITIAL VALUE)
C
      READ (11,10900) CASE
      READ (11,11000) CLNCR,LIMIT
      READ (11,11100) FHM,NW,IOPT1,IOPT2,IOPT3,IOPT4,IOPT5,IOPT6,IPRT,
     +IPRT4
      IF (IPRT4 .LE. 0) IPRT4=1
C
C               FOLLOWING SEQUENCE SETS CUMUL NR ACFT DEPLOYED(EACH DAY)
C
      READ (11,11200) NACDEP
      READ (11,11200) (IDAY(I),I=1,NACDEP)
      READ (11,11200) (NAC(I),I=1,NACDEP)
      DO 800 I=1,NACDEP
        K1=IDAY(I)
        K2=IDAY(I+1)-1
        IF (I.EQ.NACDEP) K2=NW
        DO 700 J=K1,K2
  700   AC(J)=NAC(I)
  800 CONTINUE
C
C               FOLLOWING SEQUENCE SETS PROGRAM FLYING HRS(EACH DAY)
C
      READ (11,11200) NFHDAY
      READ (11,11200) (IDAY(I),I=1,NFHDAY)
      READ (11,11200) (NFH(I),I=1,NFHDAY)
      DO 1000 I=1,NFHDAY
        K1=IDAY(I)
        K2=IDAY(I+1)-1
        IF (I.EQ.NFHDAY) K2=NW
        DO 900 J=K1,K2
  900   FHR(J)=NFH(I)
 1000 CONTINUE
C
C               FOLLOWING SEQUENCE SETS NR ACFT LOST(ATTRITION)ON EACH DAY
C
      READ (11,11200) NLDAY
      READ (11,11200) (IDAY(I),I=1,NLDAY)
      READ (11,11300) (ZLOSS(I),I=1,NLDAY)
      DO 1200 I=1,NLDAY
        K1=IDAY(I)
        K2=IDAY(I+1)-1
        IF (I.EQ.NLDAY) K2=NW
        DO 1100 J=K1,K2
 1100   ALR(J)=ZLOSS(I)
 1200 CONTINUE
C
C               FOLLOWING SEQUENCE SETS INPUT  MIN ACFT AVAIL REQMT FOR
C               EACH DAY
C
      READ (11,11200) NMDAY
      READ (11,11200) (IDAY(I),I=1,NMDAY)
      READ (11,11400) (AM(I),I=1,NMDAY)
      DO 1400 I=1,NMDAY
        K1=IDAY(I)
        K2=IDAY(I+1)-1
        IF (I.EQ.NMDAY) K2=NW
        DO 1300 J=K1,K2
 1300   DMD(J)=-AM(I)
 1400 CONTINUE
C
C               DETERMINE MIN INPUT DAILY ACFT AVAIL RQMT(OVER ALL DAYS)
C
      IK=MAXC(NW)
      XAV=-DMD(IK)
```

```
      DO 1500 I=1,NW
 1500 AVM(I)=-DMO(I)
C
C READ PART NRS(SUBSCRIPTS) OF 5 ITEMS SELECTED FOR OUTPUT OF
C CUMULATIVE TOTAL(INIT STK=0) UNCONSTR COST RQMT ON EACH DAY
      READ (11,11200) IMSEL
      READ (11,11200) (IPT(K),K=1,5)
      WRITE (6,10200) CASE
      WRITE (6,10300)
      WRITE (6,12200)
      WRITE (6,10500)
C
C PRINT INPUT-ORDERED LIST CONTAINING RANK,PART NR(FROM INPUT),
C NSN(STK NR OF PART),DESCRIPTION, UNIT COST AND INIT STOCK OF PART
C
      DO 1510 K=1,NP
        IF (MOD(K,5)).NE.0) GO TO 1510
        WRITE (6,10200) CASE
        WRITE (6,10300)
        WRITE (6,12200)
        WRITE (6,10500)
 1510 WRITE (6,12400) K,K,AMSN(K),ADESC(K),COST(K),ISTK(K)
C
C PRINT SCENARIO INPUT  DATA BASE SUMMARY IF IPRT .NE. 0
C
      IF (IPRT  .EQ. 0)GO TO 1545
      DO 1540 J=1,NW
      IF (MOD(J,5)) .NE. 0  .AND. J .GT. 1)GO TO 1535
        WRITE (6,10200) CASE
      WRITE (6,10210)
      WRITE (6,10214) ADDOST,CONVF,LIMIT,IESS
      WRITE (6,10215) FHM,CLNCR,IPRT4
      WRITE (6,10220)
      WRITE (6,10222)
 1535 CALR=CALR+ALR(J)
 1540 WRITE  (6,10230) J,AC(J),FHR(J),AVM(J),ALR(J),CALR
 1545 DO 1600 I=1,NP
 1600 STK(I)=ISTK(I)
C
C                CALCULATE TOTAL VALUE OF CURR INV (FOR PARTS PROCESSED)
C
      ZCOST=0.
      DO 1900 I=1,NP
 1900 ZCOST=ZCOST+STK(I)*COST(I)
      WRITE (6,10200) CASE
      WRITE (6,11900)
      WRITE (6,12000) ZCOST
      M1=IPT(1)
      M2=IPT(2)
      M3=IPT(3)
      M4=IPT(4)
      M5=IPT(5)
      DO 2000 I=1,NW
        DCOSTF(I)=0.
        DCOSTZ(I)=0.
        ECOSTN(I)=0.
        ECOSTZ(I)=0.
        ECOSTF(I)=0.
 2000 DCOSTN(I)=0.
      DO 2100 I=1,NP
 2100 DMO(I)=COST(I)
C
C ORDER PARTS BY DECR UNIT COST,SO IRC(1) IS PART NR(BASED ON INPUT
C ORDER ) OF MOST EXPENSIVE ITEM
C
      DO 2200 K=1,NP
        IRC(K)=MAXC(NP)
        II=IRC(K)
 2200 DMO(II)=-1.
      WRITE (6,10200) CASE
      WRITE (6,12100)
      WRITE (6,12200)
      WRITE (6,10500)
C
C PRINT COST-RANKED LIST CONTAINING RANK,PART NR(FROM INPUT),
C NSN(STK NR OF PART),DESCRIPTION OF PART, AND UNIT COST OF PART
C
      DO 2400 K=1,NP
        IF (MOD(K,5)).NE.0) GO TO 2300
        WRITE (6,10200) CASE
        WRITE (6,12100)
        WRITE (6,12200)
```

```
          WRITE (6,10500)
 2300     II=IRC(K)
 2400  WRITE (6,12400) K,II,AMSN(II),ADESC(II),COST(II),ISTK(I)
       WRITE (6,10200) CASE
C
C               CALC MAX NR OF NMCS ACFT ALLOWED EACH DAY AND SOURCE OF
C               THIS LIMIT
C
       CALR=0.
       DO 2700 I=1,NW
          ASURV(I)=AC(I)-CALR
          XX=AMAX1(0.,ASURV(I)*(1.-AVM(I)))
          YY=AMAX1(0.,ASURV(I)-FHP(I)/FHM)
          ALLOWB(I)=AMIN1(XX,YY)
          IF (ALLOWB(I).EQ.YY) RAV(I)='  FLYING HR PROG'
          IF (ALLOWB(I).EQ.XX) RAV(I)=' AVAIL CONSTRAIN'
 2700  CONTINUE
       TTFH=0.000001
C
C               CALC TOTAL FLYING HRS IN FULL WAR PROGRAM
C
       DO 2800 I=1,NW
 2800  TTFH=TTFH+FHR(I)
       WRITE (6,12500) TTFH
C
C THRU STMT 3100 CALC 'FULL SUB' UNCONSTR COST RQMNTS(RFCS(J),RFCS1(J))
C FOR TOTAL(INIT STK=0)RQMNTS AND RESIDUAL(INIT STK=CURR INV)RQMNTS
C FOR ALL PARTS
C
       DO 3100 J=1,NP
          CF(J)=FR(J)*QPA(J)
          BF(J)=(1.-BC(J))*(1.-ZNRT(J))*CF(J)
          DF(J)=(1.-DC(J))*(ZNRT(J))*CF(J)
          CDMD=0.
          SRMAX2(J)=-999.
          SRMAX1(J)=-999.
          DO 3000 I=1,NW
             CDMD=SR(I,J,FHR,CDMD)
             XXX=CDMD-ALLOWB(I)*QPA(J)
             SR1=XXX-STK(J)
             IF (XXX.GE.SRMAX1(J)) SRMAX1(J)=XXX
             IF (SR1.GE.SRMAX2(J)) SRMAX2(J)=SR1
             X1=AMAX1(0.,SRMAX1(J))
C
C CALC 'COST OF CUMUL (FULL SUB)RQMNTS THRU DAY I'FOR TOTAL(DCOSTF(I))
C AND FOR RESIDUAL RQMNT (ECOSTF(I)) CASES
C
             DCOSTF(I)=DCOSTF(I)+X1*COST(J)
             ECOSTF(I)=ECOSTF(I)+AMAX1(0.,X1-STK(J))*COST(J)
C
C CALC DAILY UNCONSTR COST (FULL SUB) RQMNT FOR PARTS SELECTED
C FOR OUTPUT OF INDIV 'CUMUL RQMNT THRU DAY I'
C
             DO 2900 M=1,IMSEL
                IF (J.NE.IPT(M)) GO TO 2900
                SM(I,M)=AMAX1(0.,SRMAX1(J))
 2900        CONTINUE
 3000     CONTINUE
          RFCS(J)=AMAX1(0.,SRMAX1(J))
          RFCS1(J)=AMAX1(0.,SRMAX2(J))
 3100  CONTINUE
C
C THRU STMT 32 CALC UNCONSTR COST 'NMCS=0' TOTAL(INIT STK=0)
C RQMNT(RNMCS(J)) AND RESIDUAL(INIT STK=CURR INV) RQMNT (RNMCS1(J))
C FOR EACH PART.   ALSO CALC 'CUMUL COST OF RQMNT THRU DAY I' FOR
C TOTAL RQMNT(DCOSTZ(I)) AND RESIDUAL RQMNT (ECOSTZ(I))
C
       DO 4000 J=1,NP
          CND=0.
          SRMAX2(J)=-999.
          SRMAX1(J)=-999.
          DO 3900 I=1,NW
             CND=SR(I,J,FHR,CND)
             SR2=CND-STK(J)
             IF (CND.GE.SRMAX1(J)) SRMAX1(J)=CND
             IF (SR2.GE.SRMAX2(J)) SRMAX2(J)=SR2
             X2=AMAX1(0.,SRMAX1(J))
             R2=AMAX1(0.,SRMAX2(J))
             DCOSTZ(I)=DCOSTZ(I)+X2*COST(J)
             ECOSTZ(I)=ECOSTZ(I)+R2*COST(J)
             DO 3800 M=1,IMSEL
                IF (J.NE.IPT(M)) GO TO 3800
```

```
                 SNM(I,M)=AMAX1(0.,SRMAX1(J))
 3600         CONTINUE
 3900       CONTINUE
            RNMCS(J)=AMAX1(0.,SRMAX1(J))
            RNMCS1(J)=AMAX1(0.,SRMAX2(J))
 4000 CONTINUE
C
C CALC DAYS FOR WHICH THE 'NO SUB' 'CUMUL COST OF ROMNT THRU DAY I'
C WILL BE COMPUTED
C
      IF (IPRT4.LE.0) IPRT4=1
      IF (IPRT4.GT.NW) IPRT4=NW
      INST=NW/IPRT4
      IST=NW-IPRT4*INST+1
C
C THRU STMT 3400 CALC UNCONSTR COST 'NO SUB' TOTAL (INIT STK=0)
C ROMNT(RNCS(J)) FOR EACH PART .   ALSO CALC 'CUMUL COST OF ROMNT
C THRU DAY I'(DCOSTN(I))
C
      DO 3200 L=1,NP
 3200 S(L)=0.
      NWT=NW+1
      DO 3400 I=IST,NWT,IPRT4
        IF (I.EQ.1) GO TO 3400
        II=I-1
        DCOSTN(II)=0
        CALL NCRNCT (II,FHR,NP)
        DO 3300 J=1,NP
          DCOSTN(II)=DCOSTN(II)+RNCS(J)*COST(J)
          DO 3300 M=1,IMSEL
            IF (J.EQ.IPT(M)) SNCM(II,M)=RNCS(J)
 3300   CONTINUE
 3400 CONTINUE
      IF (MOD(NW,IPRT4) .EQ. 0)GO TO 3310
      CALL  NCRNCT(NW,FHR,NP)
        DO 3320 J=1,NP
          DCOSTN(NW)=DCOSTN(NW)+RNCS(J)*COST(J)
          DO 3320 M=1,IMSEL
            IF (J.EQ.IPT(M)) SNCM(II,M)=RNCS(J)
 3320   CONTINUE
C
C THRU STMT 3700 CALC UNCONSTR COST 'NO SUB' RESIDUAL(INIT STK=CURR INV)
C ROMNT(RNCS1(J)) FOR EACH PART .    ALSO CALC 'CUPUL COST OF ROMNT
C THRU DAY I'(ECOSTN(I))
C
 3310 DO 3500 L=1,NP
 3500 S(L)=STK(L)
      DO 3700 I=IST,NWT,IPRT4
        IF (I.EQ.1) GO TO 3700
        II=I-1
        ECOSTN(II)=0
        CALL NCRNCR (II,FHR,NP)
        DO 3600 J=1,NP
          ECOSTN(II)=ECOSTN(II)+RNCS1(J)*COST(J)
 3600   CONTINUE
 3700 CONTINUE
      IF (MOD(NW,IPRT4) .EQ. 0)GO TO 3410
      CALL NCRNCR(NW,FHR,NP)
        DO 3420 J=1,NP
 3420     ECOSTN(NW)=ECOSTN(NW)+RNCS1(J)*COST(J)
 3410 WRITE (6,10200) CASE
      TCNMCS=0.
      TCFCS=0.
      TCNCS=0.
      UCNCS=0.
      UCFCS=0.
      UCNMCS=0.
C
C FOR EACH PART UNDER EACH POLICY WITH UNCONSTR COST,CALC TOTAL COST
C OF THAT PART(TYPE) IN THE TOTAL ROMT SOLUTIONS AND IN THE RESIDUAL
C REQMNT SOLUTIONS.
C THEN CALC TOTAL COST ,OVER ALL PARTS,OF EACH SOLUTION UNDER EACH
C POLICY.
C
      DO 4100 J=1,NP
        CNMCS(J)=COST(J)*RNMCS(J)
        CNMCS1(J)=COST(J)*RNMCS1(J)
        CFCS(J)=COST(J)*RFCS(J)
        CFCS1(J)=COST(J)*RFCS1(J)
        TCNMCS=TCNMCS+CNMCS(J)
        UCNMCS=UCNMCS+CNMCS1(J)
        UCFCS=UCFCS+CFCS1(J)
```

```
                 TCFCS=TCFCS+CFCS(J)
                 CNCS(J)=COST(J)*RNCS(J)
                 CNCS1(J)=COST(J)*RNCS1(J)
                 UCNCS=UCNCS+CNCS1(J)
      4100 TCNCS=TCNCS+CNCS(J)
     C
     C PRINT TOTAL COST(OVER ALL PARTS) OF THE TOTAL(INIT STK=0) RQMNT
     C SOLUTION FOR EACH POLICY
     C
           WRITE (6,10200) CASE
           WRITE (6,12600)
           WRITE (6,12700)
           WRITE (6,12800) TCNCS
           WRITE (6,12900) TCFCS
           WRITE (6,13000) TCNMCS
     C
     C PRINT TOTAL COST(OVER ALL PARTS) OF THE RESIDUAL(INIT STK=CURR INV )
     C RQMNT SOLUTION FOR EACH POLICY
     C
           WRITE (6,10200) CASE
           WRITE (6,13100)
           WRITE (6,12700)
           WRITE (6,12800) UCNCS
           WRITE (6,12900) UCFCS
           WRITE (6,13000) UCNMCS
     C
     C CALC COST OF REQUIRED CURRENT INVENTORY,I.E.THE VALUE OF INVENTORY
     C EXCLUDING STOCKAGE IN EXCESS OF THE 'NMCS=0' RQMNT FOR EACH PART.
     C
           SCOST=0.
           DO 4200 I=1,NP
             TSTK=AMIN1(STK(I),RNMCS(I))
      4200 SCOST=SCOST+TSTK*COST(I)
     C
     C CALC CLNC=THE VALUE OF 'REFUNDED' CURR INV + THE INPUT COST LIMIT.
     C THIS NUMBER BECOMES THE COST LIMIT USED TO COMPUTE 'TOTAL(INIT STK=0)
     C RQMNTS' IN THE CONSTRAINED COST CASE.
     C
           CLNC=CLNCR+ZCOST
     C
     C PRINT SUMMARY OF CURRENT INVENTORY VALUE AND COST LIMITS FOR THE
     C CONSTRAINED COST TOTAL RQMNT AND RESIDUAL(INIT STK=CUR INV) RQMNT CASE
     C
           WRITE (6,11900)
           WRITE (6,13200)
           WRITE (6,13300) CLNCR
           WRITE (6,13400) SCOST
           WRITE (6,13500) CLNC
           IF (IOPT1.EQ.0) GO TO 4500
     C
     C PRINT LIST (IN ORDER OF DECR PART UNIT COST)OF TOTAL(INIT STK=0)
     C UNCONSTRAINED COST RQMNTS FOR EACH PART UNDER EACH POLICY.
     C ALSO CALC FRAC OF TOTAL RQMNT(ALL PARTS) REPRESENTED BY EACH PART.
     C
           DO 4400 I=1,NP
             II=IRC(I)
             IF (MOD(I-1,50).NE.0) GO TO 4300
             WRITE (6,10200) CASE
             WRITE (6,13600) XAV
             WRITE (6,10500)
             WRITE (6,13700)
             WRITE (6,10500)
             WRITE (6,13800)
             WRITE (6,10500)
      4300   TC=100.*CNCS(II)/(TCNCS+.000001)
             TA=100.*CNMCS(II)/(TCNMCS+.000001)
             TB=100.*CFCS(II)/(TCFCS+.000001)
      4400 WRITE (6,13900) AMSN(II),ADESC(II),FFCS(II),CFCS(II),TB,RNCS(II),
          +CNCS(II),TC,RNMCS(II),CNMCS(II),TA
      4500 IF (IOPT2.EQ.0) GO TO 4900
     C
     C              PRINT LIST (IN ORDER OF DECR PART UNIT COST)OF RESIDUAL
     C              UNCONSTRAINED COST RQMNTS FOR EACH PART UNDER EACH POLICY.
     C              ALSO CALC FRAC OF TOTAL RQMNT(ALL PARTS) REPRESENTED BY
     C              EACH PART.
     C
           DO 4700 I=1,NP
             II=IRC(I)
             IF (MOD(I-1,50).NE.0) GO TO 4600
             WRITE (6,10200) CASE
             WRITE (6,14000) XAV
             WRITE (6,10500)
```

```
            WRITE (6,13700)
            WRITE (6,10500)
            WRITE (6,13800)
            WRITE (6,10500)
 4600       TC=100.*CNCS1(II)/(UCNCS+.000001)
            TA=100.*CNMCS1(II)/(UCNMCS+.000001)
            TB=100.*CFCS1(II)/(UCFCS+.000001)
 4700 WRITE (6,13900) AMSN(II),ADESC(II),RFCS1(II),CFCS1(II),TB,
     +RNCS1(II),CNCS1(II),TC,RNMCS1(II),CNMCS1(II),TA
C
C WRITE ONTO UNIT 12  A  'CURRENT INVENTORY' FOR  EACH PART BASED
C ON THE RESIDUAL 'NO SUB' RQMNT BEING 'BOUGHT' AND ADDED TO CURR INV
C
      DO 4800 J=1,NP
 4800 INC(J)=RNCS1(J)+STK(J)+.5
      WRITE (12,11200) (INC(J),J=1,NP)
 4900 KNT=1
 5000 DO 5200 L=1,NW
         DO 5100 M=1,3
 5100    FHPAPD(M,L)=0.
 5200 SUMB(L)=0.
      DO 5300 I=1,6
 5300 AVAVG(I)=0.
      DO 5400 J=1,NP
 5400 DMD(J)=0.
      TSURV=0.
      IND=0
C
C THRU STMT 6100 CALC THE DAILY ACFT AVAIL AND FLY HRS/ACFT/DAY RESULTING
C WHEN THE 'FULL SUB' AND THE 'NO SUB' UNCONSTR COST RQMNTS ARE 'BOUGHT'
C AND STOCKED. THESE STMTS WILL BE EXECUTED TWICE,ONCE(KNT=1) FOR
C THE 'TOTAL(INIT STK=0) RQMNT ' CASE AND ONCE FOR THE RESIDUAL
C (INIT STK=CURR INV) CASE. FOR CAPABILITY ASSESSMENT, RESIDUAL RQMNTS
C ARE ADDED TO CURRENT INVENTORY.
C
      TAV=0.
      TAV1=0.
      DO 6100 I=1,NW
         BMAX=0.
         DO 6000 K=1,NP
            II=IRC(K)
            X=DMD(II)
            DMD(II)=SR(I,II,FHR,X)
            ZP=RNCS(II)
            IF (KNT.EQ.2) ZP=RNCS1(II)+STK(II)
            SUMB(I)=SUMB(I)+AMAX1(0.,DMD(II)-ZP)
            AUNCS=ASURV(I)-SUMB(I)
            IF (AUNCS.LT..001) AUNCS=0.
            FHPAPD(2,I)=AMIN1(FHM,FHR(I)/(AUNCS+.01))
            ZZ=AUNCS
            AUNCS=AUNCS/(ASURV(I)+.00001)
            ZP=RFCS(II)
            IF (KNT.EQ.2) ZP=RFCS1(II)+STK(II)
            BOFCS=(DMD(II)-ZP)/QPA(II)
            IF (BOFCS.LE.0.) BOFCS=0.
            ZMAX=AMAX1(BMAX,BOFCS)
            IF (ZMAX.GT.BMAX) BMAX=ZMAX
            AUFCS=ASURV(I)-BMAX
            IF (AUFCS.LT.0.) AUFCS=0.
            FHPAPD(1,I)=AMIN1(FHM,FHR(I)/(AUFCS+.01))
            ZZ=AUFCS
            AUFCS=AUFCS/(ASURV(I)+.00001)
            Z=ALLOWB(I)*QPA(II)
 6000    CONTINUE
         RNC(I)=AUNCS
         RFC(I)=AUFCS
      TAV=TAV+RFC(I)*ASURV(I)
      TAV1=TAV1+RNC(I)*ASURV(I)
 6100 CONTINUE
C
C PRINT THE DAILY ACFT AVAIL AND FLY HRS/ACFT/DAY RESULTING
C WHEN THE 'FULL SUB' AND THE 'NO SUB' UNCONSTR COST RQMNTS ARE 'BOUGHT'
C AND STOCKED.
C
      DO 6300 I=1,NW
         AX=1.-(ALLOWB(I)/(ASURV(I)+.0000001))
         IF (MOD(I-1,50).NE.0) GO TO 6200
         WRITE (6,10200) CASE
         WRITE (6,14700)
         IF (KNT.EQ.1) WRITE (6,14100)
         IF (KNT.EQ.2) WRITE (6,14200)
         WRITE (6,10500)
```

```
            WRITE (6,14800)
            WRITE (6,10500)
            WRITE (6,14900)
            WRITE (6,15000)
            WRITE (6,10500)
  6200      TSURV=TSURV+ASURV(I)
            AVAVG(1)=AVAVG(1)+RFC(I)*ASURV(I)
            AVAVG(2)=AVAVG(2)+RNC(I)*ASURV(I)
            AVAVG(3)=AVAVG(3)+AX*ASURV(I)
            AVAVG(4)=AVAVG(4)+FHPAPD(1,I)*RFC(I)*ASURV(I)/(TAV+.0001)
            AVAVG(5)=AVAVG(5)+FHPAPD(2,I)*RNC(I)*ASURV(I)/(TAV1+.0001)
  6300 WRITE (6,15100) I,RFC(I),RNC(I),AX,RAV(I),AVM(I),FHPAPD(1,I),
     +FHPAPD(2,I),I
       DO 6400 K=1,3
  6400 AVAVG(K)=AVAVG(K)/TSURV
       WRITE (6,15200) (AVAVG(K),K=1,5)
       KNT=KNT+1
       IF (KNT.LE.2) GO TO 5000
       IF (IOPT5.EQ.0) GO TO 6600
C
C  PRINT 'CUMULATIVE TOTAL(INIT STK=0) RQMNTS THRU DAY I' FOR THE
C  5 SELECTED PARTS (SEE IPT(J))
C
       DO 6500 I=1,NW
          IF (MOD(I-1,50).NE.0) GO TO 6500
          WRITE (6,10200) CASE
          WRITE (6,15300)
          WRITE (6,14100)
          WRITE (6,10500)
          WRITE (6,15400) AMSN(M1),AMSN(M2),AMSN(M3),AMSN(M4),AMSN(M5)
          WRITE (6,15400) ADESC(M1),ADESC(M2),ADESC(M3),ADESC(M4),ADESC(M5
     +    )
          WRITE (6,15500)
          WRITE (6,10500)
  6500 WRITE (6,15600) I,(SM(I,K),SNCM(I,K),SNM(I,K),K=1,5)
C
C  PRINT 'CUMULATIVE TOTAL(INIT STK=0) RQMNTS THRU DAY I' (FULL RQMNT)
C  FOR EACH POLICY
C
  6600 DO 6700 I=1,NW
          IF (MOD(I-1,50).NE.0) GO TO 6700
          WRITE (6,10200) CASE
          WRITE (6,15700)
          WRITE (6,10500)
          WRITE (6,15800)
  6700 WRITE (6,15900) I,DCOSTF(I),DCOSTN(I),DCOSTZ(I)
C
C  PRINT 'CUMULATIVE RESIDUAL(INIT STK=CURR INV) RQMNTS THRU DAY I'
C  (FULL RQMNT) FOR EACH POLICY
C
       DO 6800 I=1,NW
          IF (MOD(I-1,50).NE.0) GO TO 6800
          WRITE (6,10200) CASE
          WRITE (6,16000)
          WRITE (6,10500)
          WRITE (6,15800)
  6800 WRITE (6,15900) I,ECOSTF(I),ECOSTN(I),ECOSTZ(I)
       IF (IOPT6.EQ.0) GO TO 17500
C
C  REST OF PROGRAM PROCESSES THE CONSTRAINED COST CASE ONLY.
C  FIRST CHECK WHETHER COST LIMIT OF 'TOTAL(INIT STK=0) RQMNT'
C  CASE(='REFUNDED' CURR INV +INPUT COST LIMIT) EXCEEDS THE COST
C  OF THE CORRESP UNCONSTR COST RQMNT SOLUTION. IF SO,DO ONLY
C  THE RESIDUAL (INIT STK=CURR INV) RQMNT CASE.
C
       KCT=0
       CNC=TCNCS-CLNC
       IF (CNC.GT.0.) GO TO 6900
       WRITE (6,10200) CASE
       WRITE (6,16100) CLNC,TCNCS
       IOPT3=0
       GO TO 7200
C
C  CALC CONSTR COST 'TOTAL RQMNT SOLUTION' BY REMOVING THE MOST
C  EXPENSIVE PARTS FROM THE UNCONSTRAINED COST TOTAL SOLUTION UNTIL THE
C  COST OF THE REMOVED ITEMS=THE COST LIMIT FOR THE 'TOTAL RQMNT' CASE.
C
  6900 DO 7100 I=1,NP
          II=IPC(I)
          CNCS(II)=RNCS(II)*COST(II)
          IF (CNCS(II).LT.CNC) GO TO 7000
          RNCS(II)=RNCS(II)-CNC/(COST(II))
```

```
            CNCS(II)=RNCS(II)*COST(II)
            GO TO 7200
  7000      RNCS(II)=0.
            CNC=CNC-CNCS(II)
            CNCS(II)=0.
 7100 CONTINUE
C
C CALC CONSTR COST 'RESIDUAL RQMNT SOLUTION' BY REMOVING THE MOST
C EXPENSIVE PARTS FROM THE UNCONSTRAINED COST RESIDUAL SOLUTION UNTIL
C COST OF THE REMOVED ITEMS=TOTAL COST OF UNCONSTRAINED COST 'NO SUB'
C
C SOLUTION (TCNCS) MINUS THE INPUT COST LIMIT (CLNCR)
C
  7200 CNC=UCNCS-CLNCR
       IF (CNC.GT.0.) GO TO 7300
       WRITE (6,10200) CASE
       WRITE (6,16200) CLNCR,UCNCS
       KCT=2
       GO TO 8300
  7300 DO 7500 I=1,NP
          II=IRC(I)
          CNCS1(II)=RNCS1(II)*COST(II)
          IF (CNCS1(II).LT.CNC) GO TO 7400
          RNCS1(II)=RNCS1(II)-CNC/(COST(II))
          CNCS1(II)=RNCS1(II)*COST(II)
          GO TO 7600
  7400    RNCS1(II)=0.
          CNC=CNC-CNCS1(II)
          CNCS1(II)=0.
  7500 CONTINUE
  7600 IF (IOPT3.EQ.0) GO TO 7900
C
C PRINT,IN ORDER OF DECR PART UNIT COST,INDIV PART RQMNTS FOR
C THE CONSTRAINED COST 'TOTAL RQMNT' CASE.
C
       DO 7800 I=1,NP
          II=IRC(I)
          IF (MOD(I-1,50).NE.0) GO TO 7700
          WRITE (6,10200) CASE
          WRITE (6,16300) CLNC
          WRITE (6,16400) CLNC
          WRITE (6,10500)
          WRITE (6,13700)
          WRITE (6,10500)
          WRITE (6,13800)
          WRITE (6,10500)
  7700    TC=100.*CNCS(II)/(CLNC+.000001)
  7800 WRITE (6,16500) AMSN(II),ADESC(II),RNCS(II),CNCS(II),TC
  7900 IF (IOPT4.EQ.0) GO TO 8200
C
C PRINT,IN ORDER OF DECR PART UNIT COST,INDIV PART RQMNTS FOR
C THE CONSTRAINED COST 'RESIDUAL RQMNT' CASE.
C
       DO 8100 I=1,NP
          II=IRC(I)
          IF (MOD(I-1,50).NE.0) GO TO 8000
          WRITE (6,10200) CASE
          WRITE (6,16600) CLNCR
          WRITE (6,16700) CLNCR
          WRITE (6,10500)
          WRITE (6,13700)
          WRITE (6,10500)
          WRITE (6,13800)
          WRITE (6,10500)
  8000    TC=100.*CNCS1(II)/(CLNCR+.000001)
  8100 WRITE (6,16500) AMSN(II),ADESC(II),RNCS1(II),CNCS1(II),TC
  8200 KNT=1
C
C               IF CONSTR COST 'TOTAL(INIT STK=0) RQMNT' RQMNT WAS NOT
C               CALCULATED
C               DO CAPABILITY ASSESSMENT FOR RESIDUAL RQMNT CASE ONLY.
C
  8300 IF (IOPT3.EQ.0) GO TO 9500
C
C THE REST OF THE PROGRAM DOES  A CAPABILITY ASSESSMENT FOR THE
C CONSTRAINED COST CASE(S).I.E. THE PROGRAM CALCULATES DAILY(&AVERAGE)
C ACFT AVAILABLE,FRACTION PROGRAM FLYING HRS FLOWN ,AND
C FLYING HRS/AVAIL ACFT/DAY GIVEN THAT
C       (1)THE CONSTR COST 'TOTAL(INIT STK=0) RQMNT' IS STOCKED
C AND (2)THE CONSTR COST RESIDUAL(UNIT STK=CURR INV) IS STOCKED
C           ALONG WITH CURR INV.
C TO GENERATE THESE,THE FOLLOWING STMTS ARE EXECUTED TWICE,FOR
```

```
C KNT=1 AND FOR KNT=2.  THE FIRST TIME(KNT=1) THE NEW 'CURR INV'  FOR
C EACH PART IS SET= THE CONSTR COST 'TOTAL RQMNT' . THE SECOND TIME,
C (KNT=2) THE NEW 'CURR INV' FOR EACH PART IS SET = OLD 'CURR INV' +
C THE CONSTR COST RESIDUAL RQMNT. IN EACH CASE ,THE NEW 'CURR INV' IS
C DENOTED BY RNCS(J).
C
 8400 AVAVG(1)=0.
      AVAVG(6)=0.
      TFHNC=0.
      IND=0
      DO 8500 I=1,NW
         SUMB(I)=0.
 8500 CONTINUE
      DO 8600 J=1,NP
         DMDT(J)=0.
 8600 DMD(J)=0.
      XX=ASURV(1)
      TAV=0.
      DO 9200 I=1,NW
         IF (I.GT.1) XX=RNC(I-1)*ASURV(I-1)+AC(I)-AC(I-1)
         FHA(I)=AMIN1(XX*FHM,FHR(I))
         INDX=0
 8700    DO 8800 K=1,NP
            XX=DMDT(K)
            DMDT(K)=SR(I,K,FHA,XX)
            ZP=DMDT(K)-RNCS(K)
            SUMB(I)=SUMB(I)+AMAX1(0.,ZP)
            AUNCS=ASURV(I)-SUMB(I)
            IF (AUNCS.LT..001) AUNCS=0.
            XX=FHR(I)
            YY=AUNCS*FHM
            FHNC(I)=AMIN1(XX,YY)
            FHPAPD(3,I)=AMIN1(FHM,FHR(I)/(AUNCS+.01))
            FHNZ(I)=FHNC(I)/(FHR(I)+.000001)
            AUNCS=AUNCS/(ASURV(I)+.00001)
 8800    CONTINUE
         Z=ABS(FHNC(I)-FHA(I))
         INDX=INDX+1
         IF (INDX.GE.LIMIT.OR.(Z/(TTFH+1.)).LE.(CONVF/NW).OR.INDX.GT.30)
     +   GO TO 9000
         FHA(I)=.5*(FHA(I)+FHNC(I))
         SUMB(I)=0.
         DO 8900 J=1,NP
 8900    DMDT(J)=DMD(J)
         GO TO 8700
 9000    TFHNC=TFHNC+FHNC(I)
         DO 9100 J=1,NP
 9100    DMD(J)=DMDT(J)
         TNCD=TNCD+Z
         RNC(I)=AUNCS
      TAV=TAV+RNC(I)*ASURV(I)
 9200 CONTINUE
      Z=100.*TNCD/(TFHNC+.001)
C
C               PRINT THE DAILY(&AVERAGE) ACFT AVAIL,FRAC PGM FLY HRS
C               FLOWN,AND
C               FLY HRS/AVAIL ACFT/DAY FOR THE CONSTR COST CAPABILITY
C               ASSESSMENT.
C
      DO 9400 I=1,NW
         AX=1.-(ALLOWB(I)/(ASURV(I)+.000001))
         IF (MOD(I-1,50).NE.0) GO TO 9300
         WRITE (6,10200) CASE
         WRITE (6,14700)
         IF (KNT.EQ.1) WRITE (6,16400) CLNC
         IF (KNT.EQ.2) WRITE (6,16700) CLNCR
         WRITE (6,10500)
         WRITE (6,16900) Z
         WRITE (6,10500)
         WRITE (6,17000)
         WRITE (6,10500)
         WRITE (6,17100)
         WRITE (6,17200)
         WRITE (6,10500)
 9300    AVAVG(1)=AVAVG(1)+RNC(I)*ASURV(I)/TSURV
         AVAVG(6)=AVAVG(6)+FHPAPD(3,I)*RNC(I)*ASURV(I)/TAV
 9400 WRITE (6,17300) I,RNC(I),AX,I,FHNZ(I),FHPAPD(3,I)
      FNC=TFHNC/TTFH
      WRITE (6,17400) AVAVG(1),AVAVG(3),FNC,AVAVG(6)
 9500 IF (KNT.EQ.2.OR.IOPT4.EQ.0.OR.KCT.EQ.2) GO TO 17500
      DO 9600 I=1,NP
 9600 RNCS(I)=STK(I)+RNCS1(I)
```

```
      KNT=KNT+1
      GO TO 8400
 9700 FORMAT (2F5.2,I5)
 9800 FORMAT (//)
 9900 FORMAT (2X,A15,F9.0,5X,F3.0,F5.0,5F3.0,I1,10X,I5)
10000 FORMAT (I2)
10100 FORMAT (A16,//////)
10200 FORMAT (1H1,30X,'CASE= ',A16)
10210 FORMAT (//,10X,'SCENARIO INPUT DATA SUMMARY')
10214 FORMAT (//,5X,'OST OFFSET=',F6.1,' DAYS    DESIRED CONVERGECE=',
     +F5.3,3X,'MAX ITERATIONS=',I3,3X,'MAX ESSENTIALITY=',I3)
10215 FORMAT (/,5X,'  MAX FLY HRS/ACFT/DAY=',F5.1,4X,'ADD-ON COST LI',
     +'MIT=',F14.0,3X,'NO SUB CUM RQMT COST CALC EA',I3,' DAYS')
10220 FORMAT (/,13X,'CUM ACFT   PROGRAM   MIN REQ    ACFT    CUM ACFT')
10222 FORMAT (7X,'DAY   DEPLOYED   FLY HRS      AVAIL     LOST',7X,
     +'LOST')
10230 FORMAT (5X,I5,F11.0,F10.0,F10.2,F8.1,F11.1)
10300 FORMAT (//,' ITEMS RANK ORDERED IN NORMAL INPUT ORDER')
10400 FORMAT (/,' PART',5X,'MSN',14X,'DESCRIPTION',7X,' COST OST FAIL',' 
     + RT NRTS  BCY  DCY  DRT BCON DCON QPA  ESS  INIT STK')
10500 FORMAT (/)
10600 FORMAT (9X,A16,2X,A16,F8.0,F3.0,F8.6,F5.2,3F5.0,2F5.2,1X,F3.0,I5,
     +I10)
10700 FORMAT (1X,I4,4X,A16,2X,A16,F8.0,F3.0,F8.6,F5.2,3F5.0,2F5.2,1X,
     +F3.0,I5,I10)
10800 FORMAT (' TOTAL NR PARTS=',I4,'   NR USED=',I4)
10900 FORMAT (1X,A16)
11000 FORMAT (1X,F14.0,I5)
11100 FORMAT (1X,F9.1,I5,5X,6I5,10X,2I5)
11200 FORMAT (16I5)
11300 FORMAT (16F5.1)
11400 FORMAT (16F5.2)
11500 FORMAT (/,5X,'CURRENT INVENTORY FOR EACH PART PROCESSED')
11600 FORMAT (/,5X,'PART NR/PART NR/NR PARTS IN TOTAL INVENTORY')
11700 FORMAT (' RANK PART   NR RANK PART   NR RANK PART   NR RANK PART''
     +    NR RANK PART   NR RANK PART   NR')
11800 FORMAT (20I5)
11900 FORMAT (///)
12000 FORMAT (/,10X,' COST OF CURRENT INVENTORY=',F14.0)
12100 FORMAT (/,' ITEMS RANK ORDERED BY  DECREASING PART COST')
12200 FORMAT (/,' RANK PART',8X,'MSN',18X,'DESCRIPTION',13X,'COST',3X,
     +'   INIT STK')
12300 FORMAT (/,5X,'COST RANKING/PART NR/STOCK NR/PART DESCRIP','/UNIT C
     +OST($)')
12400 FORMAT (2I5,5X,A16,5X,A16,2X,F14.0,4X,I8)
12500 FORMAT (2X,'          ',F10.1)
12600 FORMAT (/,10X,'TOTAL(INIT STK= 0) COST OF POLICIES')
12700 FORMAT (/,'  POLICY             TOT COST')
12800 FORMAT (/,'    NO  SUB',F14.0)
12900 FORMAT ('  FULL SUB ',F14.0)
13000 FORMAT ('  NMCS=0   ',F14.0)
13100 FORMAT (/,10X,'RESIDUAL(INIT STK=CURR STK) COST OF POLICIES')
13200 FORMAT (42X,'NO SUBST ')
13300 FORMAT (/,7X,' COST CONSTRAINT OF ADDED BUY=',F14.0)
13400 FORMAT (/,1X,' COST OF CURRENT RECUIRED INVENTORY=',F14.0)
13500 FORMAT (/,3X,' TOTAL (CURR INV+ADDED  BUY) COST=',F14.0)
13600 FORMAT (/,' TOTAL(INIT STK=0)  STK RQMTS BY POLICY ** MINIMUM AC A
     +VAIL=',F5.2,' ** PARTS IN ORDER OF DECREASING UNIT COST')
13700 FORMAT (44X,'FULL  SUBST',19X,'NO  SUBST ',23X,'NMCS=0')
13800 FORMAT (14X,'PART',21X,'RQMNT',7X,'COST  %COST',7X,'RQMNT',7X,'COS
     +T  %COST',7X,'RQMNT',7X,'COST  %COST')
13900 FORMAT (2X,A16,2X,A16,3(F8.1,F12.0,F6.2,4X))
14000 FORMAT (/,' RESIDUAL(INIT STK=CURR STK) STK RQMT BY POLICY *MINI',
     +'MUM AC AVAIL=',F5.2,' ** PARTS IN ORDER OF DECREASING UNIT COST')
14100 FORMAT (//,30X,'*** CASES ASSUME TOTAL(INIT STK=0) RECMTS',' ARE S
     +TOCKED ***')
14200 FORMAT (//,30X,'*** CASES ASSUME RESIDUAL(INIT STK=CURR STK)',' RE
     +QMTS  ARE STOCKED ***')
14300 FORMAT (/,22X,' CRIT',8X,' AC',6X,'AC',27X,'ALLOWED  NET',23X,'  P
     +ART')
14400 FORMAT ('  POLICY   DAY ',8X,'PART '8X,'DEPLOY  SURVIV NMCS AC',' 
     +AC UP AC OA BKORDERS DEMANDS',23X,'  NR')
14500 FORMAT (' NO SUBST ',I5,4X,A16,F7.0,3F8.0,F6.3,F8.1,F8.1,6X,A16,I6
     +)
14600 FORMAT (' FULL SUB ',I4,3X,A16,F7.0,3F8.0,F6.3,F8.0,F8.1,6X,A16,I6
     +)
14700 FORMAT (/,30X,'** FORCE  CAPABILITY GIVEN THAT THE COMPUTED',' REQ
     +UIREMENT (FOR EACH POLICY) IS STOCKED **')
14800 FORMAT (/,9X,'AIRCRAFT AVAILABILITY',36X,'FLY HRS / ACFT / DAY')
14900 FORMAT (14X,'FULL',9X,'NO',38X,'FULL',8X,'NO')
15000 FORMAT (6X,'DAY',6X,'SUB',8X,'SUB','  REQ AVAIL   AVAIL',' SOURCE'
     +,' AVAIL',7X,'SUB',7X,'SUB',5X,'DAY')
```

```
15100 FORMAT (5X,I4,4X,F5.3,6X,F5.3,6X,F5.3,A16,F5.2,2F10.1,I8)
15200 FORMAT (/,1X,'    AVERAGE= ',F5.3,6X,F5.3,6X,F5.3,21X,2F10.1)
15300 FORMAT (/,42X,'CUM STOCK REQUIRED THROUGH GIVEN DAY')
15400 FORMAT (8X,5(A16,8X))
15500 FORMAT (3X,'DAY FULL SB NO  SUB  NMCS=0 FULL SB NO  SUB',' NMCS=0
     + FULL SB NO  SUB  NMCS=0 FULL SB NO  SUB  NMCS=0',' FULL SB NO  SU
     +B  NMCS=0')
15600 FORMAT (2X,I4,15F8.1)
15700 FORMAT (/,12X,'CUM TOTAL(INIT STK=0) COST OF REQ THRU GIVEN DAY')
15800 FORMAT (/,6X,'DAY',3X,'   FULL SUB    NO  SUB     NMCS=0')
15900 FORMAT (6X,I3,3X,3F11.0)
16000 FORMAT (/,12X,'CUM RESIDUAL(INIT STK= CURR STK) COST OF REQ THRU',
     +' GIVEN DAY')
16100 FORMAT (//,10X,'TOTAL(CURRINV+ADDED BUY) COST CONSTRAINT=',F12.0,'
     +   EXCEEDS UNCONSTRAINED POLICY COST=',F12.0)
16200 FORMAT (//,10X,'ADDED BUY CONSTRAINT=',F12.0,'  EXCEEDS RESIDUAL R
     +EQMT COST=',F12.0)
16300 FORMAT (/,10X,'TOTAL(INIT STK=0) STK REQ BY POLICY **FULL REQ COS'
     +,'T LIMIT =',F12.0,' **   PARTS IN ORDER OF DECR UNIT COST')
16400 FORMAT (//,30X,'** TOTAL (INIT STK=0)COST OF *CURR INVENTORY','+ A
     +DDED BUY* (=',F14.0,' )   AVAIL FOR REALLOCATION')
16500 FORMAT (2X,A16,2X,A16,30X,F8.1,F12.0,F6.2,4X)
16600 FORMAT (/,10X,'RESIDUAL(INIT STK=CURR STK STK REQ  ** ADD-ON COST'
     +,' LIMIT =',F12.0,' **   PARTS IN ORDER OF DECR UNIT COST')
16700 FORMAT (//,30X,'** CURR STK=INIT STK, ONLY COST OF ADDED BUY (=',F
     +14.0,' )   IS AVAIL FOR REALLOCATION')
16900 FORMAT (' TOTAL  FLY HRS CONVERGED TO',' WITHIN',F7.3,' PERCENT')
17000 FORMAT (9X,'AIRCRAFT AVAILABILITY',41X,' FRAC FLY HR ACH',4X,'   F
     +LY HR/AC/DAY/')
17100 FORMAT (27X,'NO',48X,'NO',20X,'NO')
17200 FORMAT (16X,'DAY',7X,'SUB',13X,'REQ AVAIL',16X,'DAY',6X,'SUB',19X,
     +'SUB')
17300 FORMAT (15X,I4,5X,F5.3,17X,F5.3,15X,I4,4X,F5.3,14X,F8.1)
17400 FORMAT (/,' AVERAGE AVAIL',10X,F5.3,17X,F5.3,13X,'FRAC FLY',' HRS
     +DONE=   ',F5.3,'  AVG FH/AC/DAY= ',F5.1)
17500 END
```

SUBROUTINE NCRNCT

```
      SUBROUTINE NCRNCT (NO,FHR,NP)
C NAME: NCRNCT             TYPE: SUBROUTINE
C
C PURPOSE: THE NCRNCT (NO CANNIBALIZATION REQUIREMENTS-TOTAL) SUBROUTINE
C GENERATES  A LEAST COST TOTAL (INIT  INV=0) RQMNTS MIX OF SPARE PARTS
C NEEDED TO MEET A FLYING HR PROGRAM USING A 'NO SUBSTITUTION' PART
C REPLACEMENT POLICY AND UNCONSTRAINED  COSTS.
C
C ARGUMENTS:
C
C          NAME      DIMENSION  TYPE           DESCRIPTION
C
C
C FHR(I)             120  REAL     SEE ARGUMENT LIST ABOVE
C
C
C          NO                 1  FIXED     INPUT ARGUMENT SET TO NR  DAYS (FROM
C                                          DAY 1) FOR WHICH FULL RQMNT IS TO BE
C                                          COMPUTED
C
C          FHR(I)           120  REAL      PROGRAM FLYING HRS REQUIRED ON DAY I
C                                          (SAME AS FHR(I) IN MAIN PROG)
C
C          NP                 1  FIXED     NR OF PART TYPES TREATED (SAME AS NP
C                                          IN MAIN PROGRAM)
C CALLED BY: MAIN PROGRAM
C
C CALLS
C      -FUNCTION SR
C
C FILES USED : NO FILES READ OR WRITTEN
C
C LOCAL ARRAYS
C
C
C NAME       DIMENSION  TYPE               DESCRIPTION
C
C
C SUMBZ(I)           120  REAL     CUMULATIVE RAW(INIT STK=0) DEMANDS
C                             (ALL PARTS) THRU DAY I
C
C
C
C COMMON BLOCK (UNLABELED) ENTRIES USED IN THIS ROUTINE
C
C
C NAME       DIMENSION  TYPE               DESCRIPTION
C
C
C ALLOWB(I)          120  REAL     MAXIMUM ALLOWABLE NMCS AC ON DAY I WHICH
C                                  WILL STILL ALLOW ACHIEVMENT OF CASE OBJECTIVE
C                                  (FLY HR AND AVAILABILITY) ON DAY I
C
C IRC(J)             300  REAL     PART NR CORRESPONDING TO THE J-TH MOST
C                                  COSTLY PART TYPE.(SEE MAIN PROGRAM COMMON)
C                                  THIS ROUTINE MUST PROCESS PARTS IN ORDER
C                                  OF DECREASING UNIT COST.
C
C RNCS(J)            300  REAL     TOTAL REQMT(INIT STK=0) FOR PART J USING A
C                                  'NO SUBSTITUTION' REPLACEMENT POLICY
C                                   WITH UNCONSTRAINED COST. THIS QUANTITY
C                                   IS CALCULATED ONLY IN THIS ROUTINE,AND
C                                   IS PASED THRU COMMON TO THE  MAIN PROG
C
C
      COMMON
     +     ALLOWB(120),    BCY(700),       BF(300),        CF(300),
     +     DCY(300),       DF(300),        DMD(300),       IRC(300),
     +     RNCS(300),      RNCSI(300),     RNMCS(300),     S(300)
      DIMENSION
     +     FHR(120),       SUMBZ(120)
      SUMR=0.
      DO 100 L=1,NO
  100 SUMBZ(L)=0.
      TSUMB=0.
C
C                PROCESS PARTS IN DECREASING COST ORDER
C
      DO 300 K=1,NP
        II=IRC(K)
C
```

```
C                INITIALIZE ROMNT TO INITIAL INVENTORY(=0)   (S(II)=0)
C
         RNCS(II)=S(II)
         CDMD=0.
         DO 200 I=1,ND
C
C   CALC CUMULATIVE NET DEMAND (CDMD) FOR PART II THRU DAY I.
C   THEN CALC (SUMBZ(I)) TOTAL NET DEMAND THRU DAY I OVER THE K MOST
C   EXPENSIVE PART TYPES. FINALLY CALC (TSUMB) THE NET TOTAL STOCKOUT
C   ('HOLES') THRU DAY I AND SET THE ROMNT FOR PART II= THE DIFFERENCE
C   BETWEEN THE NET TOTAL STOCKOUT AND THE ALLOWABLE STOCKOUT(ALLOWB(I)).
C   THIS CALC IMPLICITLY ASSUMES (THRU SUMR) THAT THE ROMNTS FOR THE
C   (K-1) MOST EXPENSIVE PARTS HAVE BEEN COMPUTED AND BOUGHT.
C
           CDMD=SR(I,II,FHR,CDMD)
           SUMBZ(I)=SUMBZ(I)+CDMD
           TSUMB=AMAX1(SUMBZ(I)-SUMR,0.)
           IF ((TSUMB-RNCS(II)).GE.ALLOWB(I)) RNCS(II)=TSUMB-ALLOWB(I)
     200   CONTINUE
C
C   CALC (SUMR) TOTAL UNITS STOCK REQUIRED FOR THE K MOST EXPENSIVE
C   PARTS
C
         SUMR=SUMR+RNCS(II)
     300 CONTINUE
        RETURN
        END
```

SUBROUTINE NCRNCR

```
      SUBROUTINE NCRNCR (ND,FHR,NP)
C NAME: NCRNCR              TYPE: SUBROUTINE
C
C PURPOSE: THE NCRNCT (NO CANNIBALIZATION REQUIREMENTS-RESIDUAL)SUBROUTINE
C GENERATES  A LEAST COST RESIDUAL(INIT  INV=CURRENT INVENTORY) RQMNTS MIX
C OF SPARE PARTS NEEDED TO MEET A FLYING HR PROGRAM USING A 'NO SUBSTITUTION
C REPLACEMENT POLICY AND UNCONSTRAINED  COSTS.
C
C ARGUMENTS:
C
C      NAME      DIMENSION  TYPE          DESCRIPTION
C
C
C FHR(I)            120  REAL     SEE ARGUMENT LIST ABOVE
C
C
C      ND                  1  FIXED     INPUT ARGUMENT SET TO NR  DAYS (FROM
C                                       DAY 1) FOR WHICH FULL RQMNT IS TO BE
C                                       COMPUTED
C
C      FHR(I)           120  REAL       PROGRAM FLYING HRS REQUIRED ON DAY I
C                                       (SAME AS FHR(I) IN MAIN PROG)
C      NP                  1  FIXED     NR OF PART TYPES TREATED (SAME AS NP
C                                       IN MAIN PROGRAM)
C CALLED BY: MAIN PROGRAM
C
C CALLS
C     -FUNCTION SR
C
C FILES USED : NO FILES READ OR WRITTEN
C
C LOCAL ARRAYS
C
C
C NAME       DIMENSION  TYPE               DESCRIPTION
C
C
C SUMBZ(I)          120  REAL     CUMULATIVE RAW(INIT STK=0) DEMANDS
C                           (ALL PARTS) THRU DAY I
C
C SUMP(I)           120  REAL     TOTAL UNITS(ALL PARTS) STOCKED IN EXCESS
C                           OF EXPECTED DEMAND ON DAY I)
C
C
C
C COMMON BLOCK (UNLABELED) ENTRIES USED IN THIS ROUTINE
C
C
C NAME       DIMENSION  TYPE               DESCRIPTION
C
C
C ALLOWB(I)         120  REAL     MAXIMUM ALLOWABLE NMCS AC ON DAY I WHICH
C                                 WILL STILL ALLOW ACHIEVMENT OF CASE OBJECTIVE
C                                 (FLY HR AND AVAILABILITY) ON DAY I
C
C IRC(J)            300  REAL     PART NR CORRESPONDING TO THE J-TH MOST
C                                 COSTLY PART TYPE.(SEE MAIN PROGRAM COMMON)
C                                 THIS ROUTINE MUST PROCESS PARTS IN ORDER
C                                 OF DECREASING UNIT COST.
C
C RNCS1(J)          300  REAL     RESIDUAL REQMT(INIT STK=CURR STK) FOR PART J
C                                 USING A 'FULL SUBSTITUTION' REPLACEMENT POLICY
C                                 WITH UNCONSTRAINED COST. THIS QUANTITY
C                                  IS CALCULATED ONLY IN THIS ROUTINE,AND
C                                  IS PASED THRU COMMON TO THE  MAIN PROG
C
C
      COMMON
     +    ALLOWB(120),    RCY(300),      BF(300),        CF(300),
     +    DCY(300),       DF(300),       DMD(300),       IRC(300),
     +    RNCS(300),      RNCS1(300),    RNMCS(300),     S(300)
      DIMENSION
     +    FHR(120),       SUMBZ(120),    SUMP(120)
      SUMR=0.
      DO 100 L=1,ND
        SUMP(L)=C.
  100 SUMBZ(L)=0.
      TSUMB=0.
C
C               PROCESS PARTS IN DECREASING COST ORDER
C
```

```
      DO 300 K=1,NP
        II=IRC(K)
C
C             INITIALIZE RESIDUAL RQMNT  TO  INITIAL INVENTORY(S(II)
C
        RNCS1(II)=S(II)
       IF (S(II) .GT. RNMCS(II)) GO TO 300
        COMD=0.
        DO 200 I=1,ND
C
C  CALC CUMULATIVE NET DEMAND (COMD) FOR PART II THRU DAY I.
C  THEN CALC EXCESS (SUMP(I) OFCURRENT INVENTORY OVER NET DEMAND
C  THEN CALC (SUMBZ(I)) TOTAL NET DEMAND THRU DAY I OVER THE K MOST
C  EXPENSIVE PART TYPES. FINALLY CALC (TSUMB) THE NET TOTAL STOCKOUT
C  ('HOLES') THRU DAY I AND SET THE RQMNT FOR PART II= THE DIFFERENCE
C  BETWEEN THE NET TOTAL STOCKOUT AND THE ALLOWABLE STOCKOUT(ALLOWB(I)).
C  THIS CALC IMPLICITLY ASSUMES (THRU SUMR) THAT THE RQMNTS FOR THE
C  (K-1) MOST EXPENSIVE PARTS HAVE BEEN COMPUTED AND BOUGHT.
C
          COMD=SR(I,II,FHR,COMD)
          SUMP(I)=SUMP(I)+AMAX1(0.,(RNCS1(II)-COMD))
          SUMBZ(I)=SUMBZ(I)+COMD
          TSUMB=AMAX1(SUMBZ(I)-SUMR+SUMP(I),0.)
          IF ((TSUMB-RNCS1(II)).GE.ALLOWB(I)) RNCS1(II)=TSUMB-ALLOWB(I)
  200   CONTINUE
C
C  CALC (SUMR) TOTAL UNITS STOCK REQUIRED FOR THE K MOST EXPENSIVE
C  PARTS
C
        SUMR=SUMR+RNCS1(II)
  300 CONTINUE
      DO 400 J=1,NP
  400 RNCSI(J)=RNCSI(J)-S(J)
      RETURN
      END
```

FUNCTION SR

```
      FUNCTION SR (I,J,FHR,COMD)
C NAME: SR                TYPE: FUNCTION
C
C PURPOSE: THE SR (STOCK REQUIRED) FUNCTION CALCULATES THE CULMULATIVE
C          NET DEMAND THRU A SPECIFDIED DAY FOR A SPECIFIED PART BASED
C          ON A SPECIFIED FLYING PROGRAM. INITIAL INVENTORY =0 IS
C          ASSUMED IN THIS CALCULATION. NET DEMANDS IS BASICALLY
C          FAILED ITEMS OFFSET BY RETURNING REPAIRS. IN A SENSE IT'S THE
C          NET NR OF 'HOLES' (CAUSED BY THE ITEM) WHICH ARE PRESENT ON
C          A SPECIFIED DAY ,ASSUMING A ZERO INITIAL INVENTORY.
C
C ARGUMENTS:
C
C      NAME      DIMENSION  TYPE          DESCRIPTION
C
C
C
C      COMD              1  REAL      THE CUMULATIVE NET DEMAND THRU THE
C                                     PREVIOUS DAY(I-1)FOR PART NR J.
C                                     IT= 0 OR A PREVIOUSLY CALCULATED VALUE
C                                     (FOR DAY I-1) OF SR.
C
C      FHR(I)          120  REAL      PROGRAM FLYING HRS FLOWN ON DAY I.
C                                     SOMETIMES THIS WILL BE THE SAME AS
C                                     FHR(I) IN THE MAIN PROG.SOMETIMES(IN
C                                     CAPABILITY ASSESSMENT) THIS WILL BE AN
C                                     ESTIMATE OF 'FLYING HRS FLOWN' ON DAY I.
C
C      I                    FIXED     CURRENT DAY
C
C      J                    FIXED     PART NR OF PART BEING PROCESSED
C
C CALLED BY: MAIN PROGRAM
C CALLS
C     -FUNCTION SR
C FILES USED : NO FILES READ OR WRITTEN
C
C LOCAL ARRAYS
C
C
C  NAME      DIMENSION  TYPE              DESCRIPTION
C
C
C  FHR(I)          120  REAL      SEE ARGUMENT LIST ABOVE
C
C
C
C COMMON BLOCK (UNLABELED) ENTRIES USED IN THIS ROUTINE
C
C
C NAME       DIMENSION  TYPE              DESCRIPTION
C
C
C
C BCY(J)           300  REAL      BASE REPAIR CYCLE TIME (DAYS) OF PART J
C                                 (=BASE REPAIR TIME OF PART)(BASE=RETAIL)
C
C BF(J)            300  REAL      A COEFFICIENT USED IN CALCULATION OF NET
C                                 DEMANDS(SR(I,J,..)) FOR PART J. IT=
C                                 (1-BC(J))*(1-ZNRT(J))*CF(J)
C
C CF(J)            300  REAL      A COEFFICIENT USED IN CALCULATION OF NET
C                                 DEMANDS(SR(I,J,..)) FOR PART J. IT=
C                                 FR(J)*QPA(J)
C
C DCY(J)           300  REAL      DEPOT RECYCLE TIME FOR PART TYPE J,I.E.
C                                 TIME BETWEEN REMOVAL AND RETURN FROM DEPOT
C                                 REPAIR. THIS = DEPOT REPAIR TIME + 2*ORDER
C                                 SHIP TIME.
C
C
       COMMON
     +     ALLOWB(120),    BCY(300),      BF(300),       CF(300),
     +     DCY(300),       DF(300),       DMD(300),      IRC(300),
     +     RNCS(300),      RNCSI(300),    RNMCS(300),    S(300)
       DIMENSION
     +     FHR(120)
C
C  CALC (ID,IP) THE DAYS ON WHICH 'ITEMS RETURNING TODAY(DAY I)
C  FROM DEPOT' FAILED.
C
```

```
      ID=I-DCY(J)
      IB=I-BCY(J)
      DRR=0.
      BRR=0.
C
C  CALC (DRR) RETURNING REPAIRS(RETURNING ON DAY I)FROM DEPOT  AND
C CALC (BRR) RETURNING REPAIRS FROM RETAIL. THEN DETERMINE CUMULATIVE
C  NET DEMAND (THRU TODAY(DAY I)) BY ADD NET DEMANDS GENERATED
C  TODAY TO CUMULATIVE NET DEMANDS THRU YESTERDAY,
C
      IF (ID.GT.0) DRR=DF(J)*FHR(ID)
      IF (IB.GT.0) BRR=BF(J)*FHR(IB)
      CR=CDMD+CF(J)*FHR(I)-DRR-BRR
      RETURN
      END
```

FUNCTION MAXC

```
      FUNCTION MAXC (NP)
C NAME: FMAX                  TYPE: FUNCTION
C
C PURPOSE: THE FMAX FUNCTION FINDS THE SUBSCRIPT OF THE LARGEST(IN VALUE
C          MEMBER OF AN ARRAY (DMD(J))
C
C ARGUMENTS:
C
C      NAME      DIMENSION  TYPE          DESCRIPTION
C
C      NP                1  FIXED     NR OF ITEMS IN ARRAY TO BE ORDERED.
C                                     USUALLY,THIS IS THE NR OF PART TYPES
C                                     PROCESSED,BUT ONCE IN THE MAIN PGM
C                                     IS THE NR OF ' MIN REQ ACFT AVAIL'
C                                     SPECIFICATIONS READ IN.
C
C CALLED BY: MAIN PROGRAM
C
C CALLS   : NONE
C
C FILES USED : NO FILES READ OR WRITTEN
C
C LOCAL ARRAYS : NONE
C
C COMMON BLOCK (UNLABELED) ENTRIES USED IN THIS ROUTINE
C
C NAME       DIMENSION  TYPE              DESCRIPTION
C
C DMD(J)           300  REAL    WORKING  ARRAY USED HERE TO STORE THE
C                               ARRAY BEING ORDERED.
C
      COMMON
     +    ALLOWB(120),    BCY(300),      BF(300),       CF(300),
     +    DCY(300),       DF(300),       DMD(300),      IRC(300),
     +    RNCS(300),      RNCS1(300),    RNMCS(300),    S(300)
      SMAX=-1.
      JMAX=1
      DO 100 J=1,NP
         X=DMD(J)
         ZMAX=AMAX1(SMAX,X)
         IF (ZMAX.LE.SMAX) GO TO 100
         JMAX=J
         SMAX=ZMAX
  100 CONTINUE
      MAXC=JMAX
      RETURN
      END
```

GLOSSARY

ABBREVIATIONS, ACRONYMS, AND SHORT TERMS

acft	aircraft
AFH	achievable flying hours
AFLC	Air Force Logistics Command
AMC	US Army Materiel Command
AR	Army regulation
ASL	authorized stockage list(s)
avail	availability
avg	average
AVIM	aviation intermediate maintenance
AVSCOM	US Army Aviation Systems Command
AVUM	aviation unit maintenance
CAA	US Army Concepts Analysis Agency
CCSS	Commodity Command Standard System
CONUS	Continental United States
cont	continued
cum	cumulative
curr	current
DCSLOG	US Army Deputy Chief of Staff for Logistics
DESCOM	US Army Depot Systems Command
DOD	Department of Defense

EFH	estimated flying hours
FH	flying hour(s)
FHP	flying hour program
hr	hour
MFHAD	maximum flying hours per aircraft per day
min	minimum
MSC	major subordinate command
NMC	not mission capable
NMCS	not mission capable due to supply
NRTS	not repairable at this station
OST	order and ship time
PARCOM	Parts Requirements and Cost Model
PLL	prescribed load list(s)
QPA	quantity per application
rqmt(s)	requirement(s)
sub	substitution

END

FILMED

4-85

DTIC